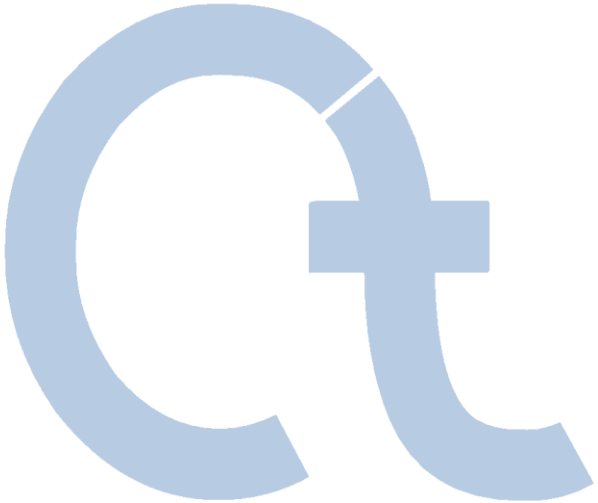


CRIPTOLOGÍA CON CRYPTOOL v 1.4.30

Introducción a la Criptografía y al Criptoanálisis

Alcance, Tecnología y Futuro de CrypTool

www.cryptool.org
www.cryptool.com
www.cryptool.de
www.cryptool.es
www.cryptool.pl

Prof. Bernhard Esslinger y el equipo de CrypTool, Agosto 2010

Contenido (I)

I. CrypTool y Criptología – Visión General

1. Definición y relevancia de la Criptología
2. El Proyecto CrypTool
3. Ejemplos de métodos clásicos de cifrado
4. Conocimientos sobre el desarrollo de la criptografía

II. Características de CrypTool

1. Visión General
2. Ejemplos de Interacción
3. Desafíos para los desarrolladores

III. Ejemplos

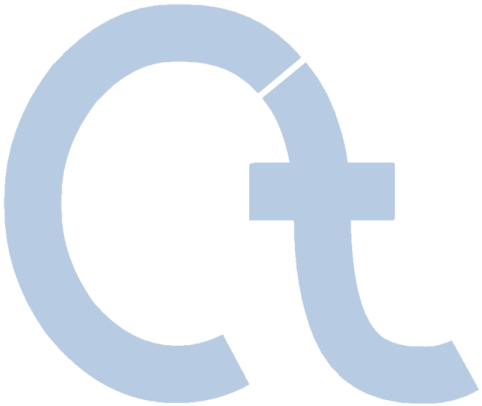
1. Cifrado con RSA / Test de primalidad / Cifrado Híbrido y certificados digitales
2. Visualización de firma digital
3. Ataque al cifrado RSA (modulo N demasiado pequeño)
4. Análisis del cifrado de la PSION 5
5. Claves DES débiles
6. Localizar información de la clave (“clave NSA”)
7. Ataque a la firma digital por localización de colisiones hash
8. Autentificación en un entorno cliente-servidor
9. Demonstración de un ataque de canal lateral(en un protocolo de cifrado híbrido) (…)

Contenido (II)

III. Ejemplos

IV. Proyecto / Perspectiva / Contacto

Contenido



Relevancia de la Criptografía

Ejemplos de Uso de la Criptografía

- Cajeros automáticos, transferencias entre bancos
- TV por Satélite, TV de pago
- Sistemas inmovilizadores en coches
- Gestión de Derechos Digitales (DRM)
- Tarjetas telefónicas, teléfonos móviles, controles remotos
- Dinero electrónico, banca electrónica, correo electrónico seguro
- La Criptografía no está limitada a las empresas, diplomacia o a los militares. La Criptografía es una caracterizada ciencia matemática.
- Un gran cambio en la criptografía empezó con la generalización del uso de Internet
- Para las empresas y los gobiernos es importante que los sistemas sean seguros y

… ¡que los usuarios (clientes, empleados) tengan un cierto entendimiento y conciencia sobre la seguridad en TI!

Definición: Criptología y Criptografía

Criptología *(del Griego kryptós, "escondido", y lógos, "palabra") es la ciencia de las comunicaciones seguras (generalmente secretas). Esta seguridad se obtiene con usuarios legítimos, el transmisor y el receptor, siendo capaz de transformar la información en un código utilizando una clave – por ejemplo, una parte de la información solamente conocida por ellos. Aunque el código es inescrutable y muy a menudo inolvidable para cualquiera con su clave secreta, el receptor autorizado podrá descifrar el código para recuperar la información escondida o verificar que fue enviado probablemente por alguien que posee la clave.*

Criptografía *al principio se preocupaba de proporcionar confidencialidad para los mensajes escritos. Sin embargo, sus leyes se aplican igualmente bien para asegurar un flujo de datos entre ordenadores o para cifrar señales televisivas. ... Hoy, las ciencias (matemáticas) modernas de criptología no sólo contienen mecanismos para cifrar sino también para la integridad, firmas electrónicas, números aleatorios, intercambio seguro de claves, recipientes seguros, voto electrónico y dinero electrónico, y también ha conseguido convertirse en una gran variedad de aplicaciones en la vida moderna.*

Fuente: Britannica (www.britannica.com)
Una definición similar se puede encontrar en Wikipedia: http://es.wikipedia.org/wiki/Criptologia

Criptografía – Objectivos

- **Confidencialidad**
  - La información prácticamente no puede ser accesible o revelada a individuos, entidades o procesos desautorizados.
- **Autentificación**
  - La autentificación asegura que los usuarios se han identificado y que sus identidades se han verificado apropiadamente.
- **Integridad**
  - La integridad asegura que los datos no se han alterado o destruido de una forma no autorizada.
- **No-Repudio**
  - El principio de que, después de todo, se puede probar que los participantes de una transacción realmente la autorizan y que no pueden negar de ninguna forma su participación.

El Proyecto CrypTool

- Origen en un programa de concienciación de un banco (Capacitación Empresarial)
  → **Concienciación para empleados**
- Desarrollado en cooperación con universidades (mejorando la educación)
  → **Enfoque didáctico y orientado a estándares**

  1998 **Inicio del proyecto** – el esfuerzo de más de 40 años-hombre desde entonces
  2000 CrypTool disponible como **software libre**
  2002 CrypTool en el **CD-ROM-Ciudadano de la BSI** (Agencia Alemana de Seguridad de la información)
  2003 CrypTool se convierte en **Código-Abierto** – Soporte por Universidad de Darmstadt (Prof. Eckert)
  2007 CrypTool disponible en alemán, inglés, español y polaco
  2008 Inicio de versiones .NET y Java – Mantenidas por la Univ. de Duisburg (Prof. Weis) y SourceForge
  2010 CT1 disponible en su quinto idioma, serbio. Preparando versiones .NET y Java para ser lanzadas

- **Galardones**

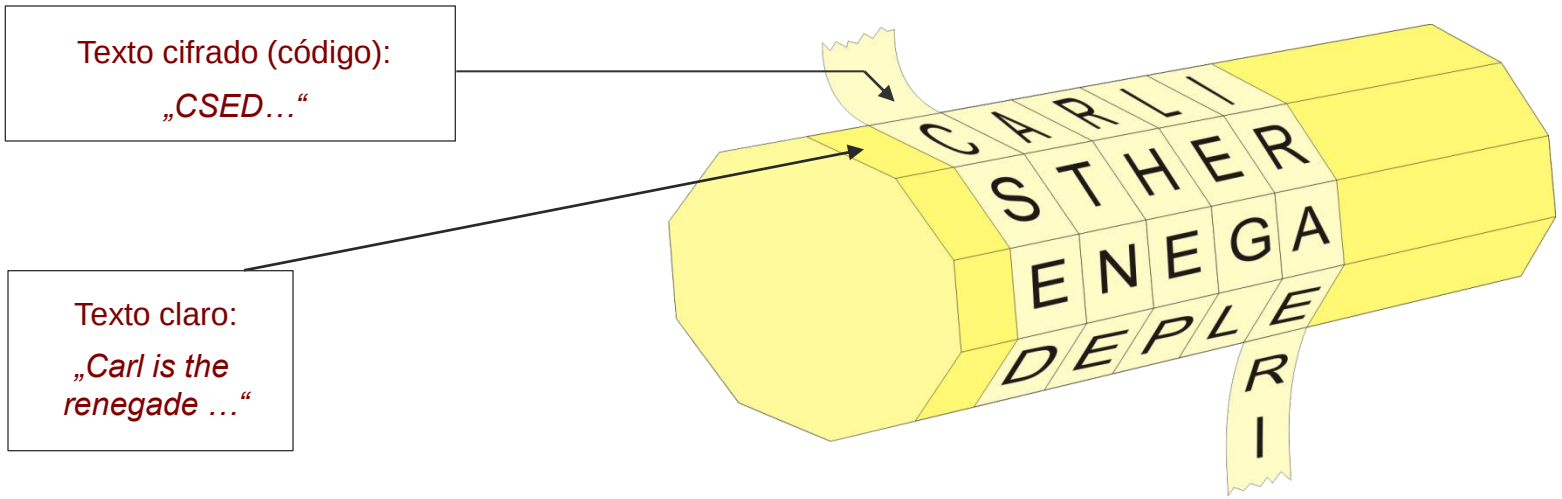
  2004 TeleTrusT (TTT Förderpreis)
  2004 NRW (IT Security Award NRW)
  2004 RSA Europe (Finalista del European Information Security Award 2004)
  2008 "Selected Landmark" en la iniciativa "Germany – Land of Ideas"

- **Desarrolladores**
  → Desarrollado por gente de empresas y universidades en distintos países
  → Miembros adicionales del proyecto o códigos útiles siempre se aprecian (actualmente existen alrededor de 50 personas trabajando sobre el universo CrypTool).

Ejemplos de la primera Criptografía (1)

Métodos de cifrado antiguos.

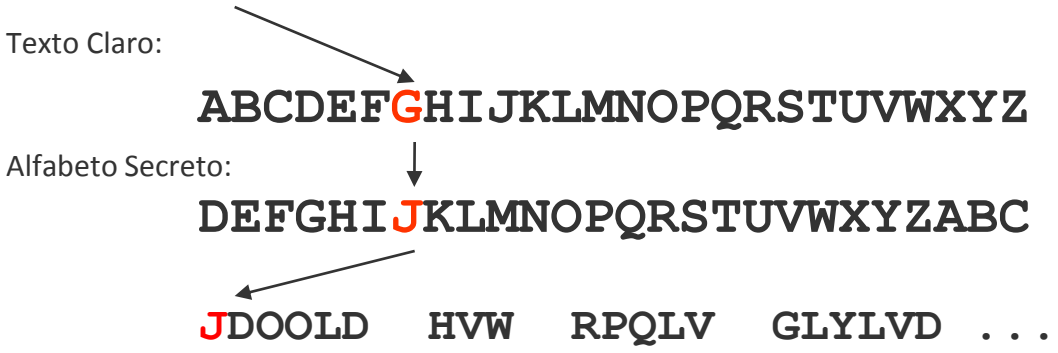
- **Tatuajes en la cabeza de un esclavo cubierto por el cabello**
- **Atbash** (sobre 600 A.C.)
  - Lenguaje secreto Hebreo, alfabeto invertido
- **Scytale de Sparta** (500 A.C.)
  - Descrito por el historiador/autor Griego Plutarco (45 - 125 A.C.)
  - Dos cilindros (varas de madera) con igual diámetro
  - Transposición (los caracteres del texto claro se reordenan)

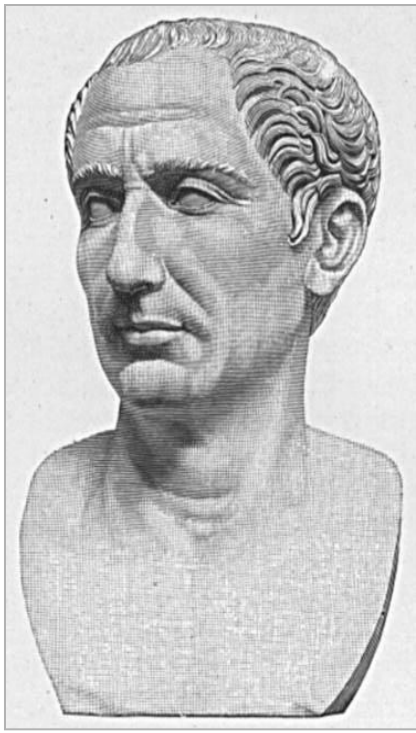


Ejemplos de la primera Criptografía (2)

Cifrado Simétrico del César

- **Cifrado César** (Julius Caesar, 100 - 44 A.C.)
- Código de sustitución simple





- **Ataque:** Análisis de frecuencias (distribución típica de caracteres)

Presentación con CrypTool mediante los siguientes menus:

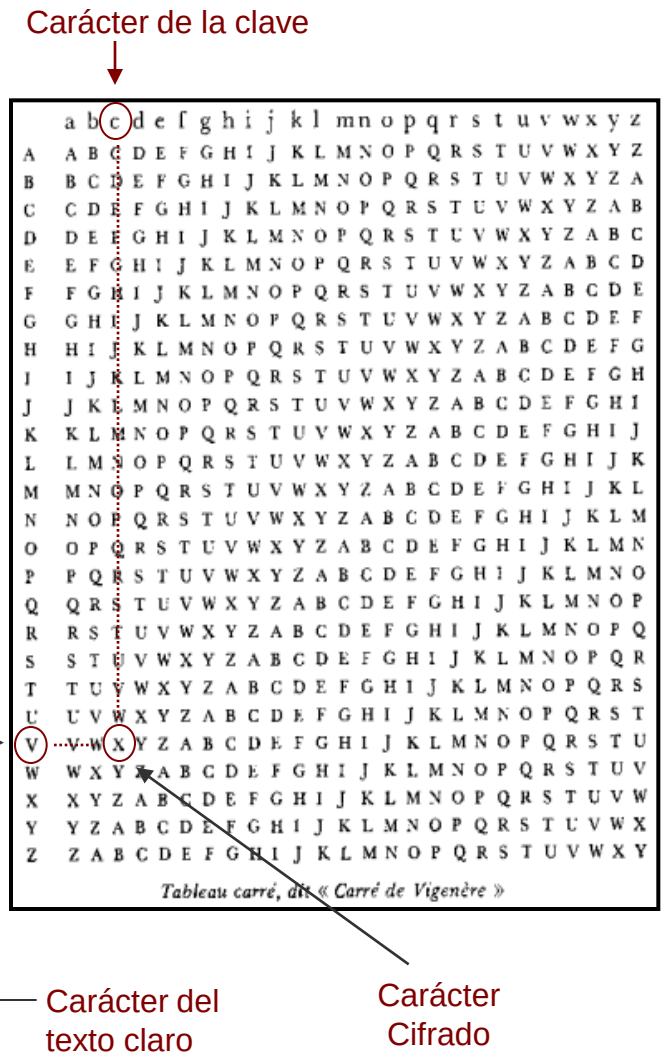
- Animación: „Procedimientos Idiv.“ \ „Visualización de algoritmos“ \ „Cesar“
- Implementación: „Cifrar/Descifrar“ \ „Simétrico (clásico)“ \ „Cesar / Rot-13“

Ejemplos de la primera Criptografía (3)

Cifrado Simétrico de Vigenère (Cifrado de sustitución polialfabética)

- **Cifrado Vigenère** (Blaise de Vigenère, 1523-1596)
- Cifrado con una palabra clave utilizando una tabla clave
- Ejemplo:
  Palabra clave: **CHIFFRE**
  Cifrando: **VIGENERE** resulta **XPOJSVVG**
- El carácter (V) del texto claro se reemplaza por el carácter en la fila correspondiente y en la columna de la primera palabra de la palabra clave (c). El siguiente carácter del texto claro (I) se reemplaza por el carácter en la fila correspondiente y en la columna de la siguiente letra de la palabra clave (h), y así sucesivamente.
- Si se han utilizado todos los caracteres de la palabra clave, entonces el siguiente carácter de la palabra clave es la primera letra de la palabra clave.
- **Ataque** (por el test Kasiski): Pueden dase combinaciones de textos claros con idénticos textos cifrado. La distancia de éstos patrones se pueden utilizar para determinar la longitud de la clave.

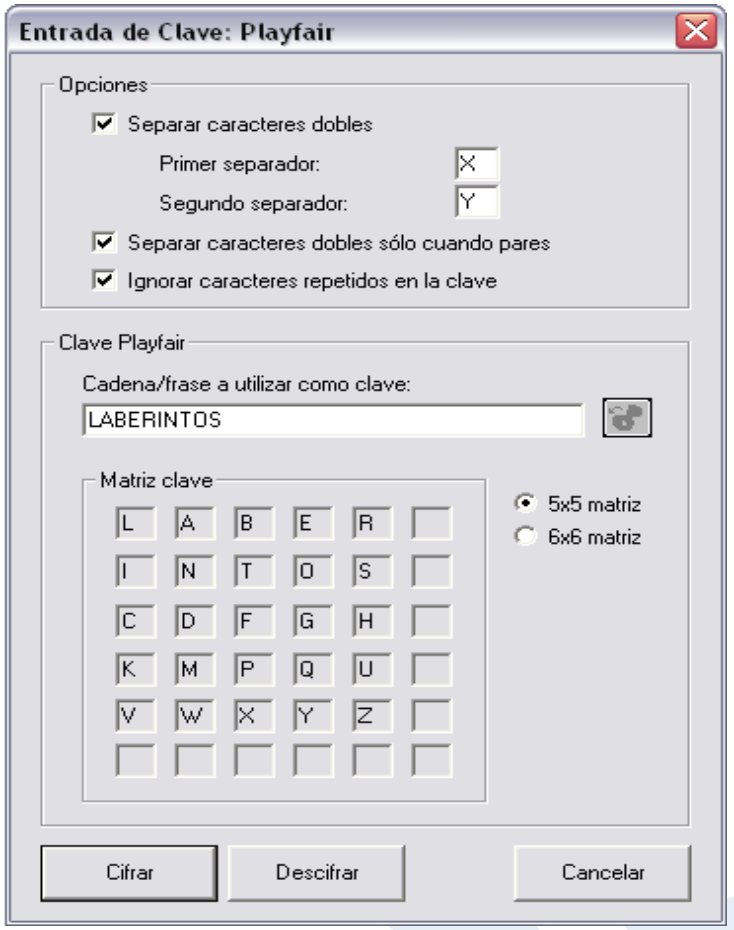
  Un análisis de frecuencia tradicional se puede utilizar para determinar la clave.



Ejemplos de la primera Criptografía (4)

Otros métodos de cifrado simétricos

- **Sustitución Homofónica**
- **Playfair** (inventado en 1854 por Sir Charles Wheatstone, 1802-1875)
  - Publicado por el Baron Lyon Playfair
  - Sustitución de un par de caracteres por otro basado en una matriz cuadrada de letras
- **Transferencia de páginas de libro**
  - Adaptación de la Libreta de un sólo uso (OTP)
- **Rejilla giratoria** (Fleissner)
- **Cifrado por permutación**
  - „Doble Dado“ (trasposición de columna doble) (Trasposición / muy efectiva)



La Criptografía en Tiempos Modernos

Desarrollo de la Criptografía en los últimos 100 años hasta 1970

Métodos Clásicos

- Todavía se utilizan actualmente .
  (ya que no todo lo puede hacer un ordenador...)
- Y sus principios de **transposición** y **sustitución** son un gran apoyo para el diseño de algoritmos modernos: la combinación de operaciones simples (un tipo de cifrado múltiple, también llamado cifrado en cascada), a nivel de bit, cifrado en bloque, ciclos.

El cifrado se vuelve

- más **sofisticado**,
- **Mecanizado** o **computarizado** y
- Permanece **simétrico**.

Ejemplos de la Primera Mitad del S. XX

Máquinas de cifrado mecánico (máquinas de rotores)

Cifrado Enigma (Arthur Scherbius, 1878-1929)

- Se han utilizado más de 200000 máquinas en la II Guerra Mundial.
- El cilindro giratorio elige las causas por las que cada carácter del texto se cifra con una nueva permutación.
- La oficina de cifrado polaca descifró el sistema Enigma prebélico ya en 1932.
- Código roto por un esfuerzo masivo por parte de expertos en criptografía (unas 7000 personas en Reino Unido) con máquinas de descifrado, Enigmas originales capturadas o interceptando comunicados de estado diarios (p.ej. comunicados meteorológicos).
- **Consecuencias de este exitoso criptoanálisis:**
  "En general, el exitoso criptoanálisis del cifrado enigma tuvo una ventaja estratégica, que jugó un papel significante para ganar la guerra. Algunos historiadores afirman que el descifrado del código enigma acortó la guerra varios meses o incluso un año."

  (traducido de http://de.wikipedia.org/wiki/Enigma_%28Maschine%29 - Marzo 6, 2006)

Criptografía – Conceptos Importantes (1)

- **Principio de Kerckhoffs** (establecido en 1883)
  - Separación del algoritmo (método) y la clave
    p.ej. Cifrado César:
    Algoritmo: "Alfabeto desplazado un cierto número de posiciones a la izquierda"
    Clave: El "cierto número de posiciones" (César por ejemplo)
  - Principio de Kerckhoffs :
    El secreto permanece en la clave y no en el algoritmo, es decir, "No hay seguridad por oscuridad"

- **Libreta de un sólo uso – Shannon / Vernam**
  - Demostrado teóricamente seguro, pero no es útil en la realidad (sólo el teléfono rojo)

- **Conceptos de Shannon : Confusión y Difusión**
  - Relación entre M, C y K tiene que ser tan compleja como sea posible (M=mensaje, C=código, K=clave)
  - Cada carácter del texto cifrado debe depender de tantos caracteres del texto claro como de la clave de cifrado.
  - „ Efecto Avalancha"(una pequeña modificación tiene un gran impacto)

- **Función de puerta trasera** (función en una dirección)
  - Rápido en una dirección pero no en la dirección contraria (sin información secreta)
  - La dirección contraria funciona teniendo el secreto (acceso a la puerta trasera)

Ejemplos de una Fisura en el Principio de Kerckhoffs

El secreto está relacionado con la clave y no con el algoritmo

- **Penetración en el cifrado de teléfonos móviles** (Diciembre 1999)

  „ Científicos Israelíes descubrieron un defecto de diseño que permitía descodificar las conversaciones privadas de cientos de millones de teléfonos móviles. Alex Biryukov y Adi Shamir describen en un artículo publicado esta semana cómo un PC con 128 MB de RAM y unos grandes discos duros puede saltarse la seguridad de una llamada telefónica o de una transmisión de datos en menos de un segundo. El algoritmo erróneo apareció en los teléfonos digitales GSM hechos por empresas como Motorola, Ericsson, y Siemens, y que son utilizados por unos 100 millones de clientes en Europa y Estados Unidos.” […]

  “Los algoritmos de cifrado GSM habían estado bajo prueba de ataques al ***estar siendo desarrollados en secreto apartados del escrutinio público –****pero muchos expertos dicen que una alta seguridad sólo puede venir de un código publicado. Moran dijo “no fue la actitud a la hora de publicar los algoritmos” cuando los códigos A5 se desarrollaron en 1989, pero* ***los actuales que se están creando se publicarán para una revisión por pares****.”*
  [http://www.wired.com/politics/law/news/1999/12/32900]

- **Otro Ejemplo:** En 1999, el navegador Netscape almacenó contraseñas para acceder al servidor de correos utilizando un método de cifrado débil.

Muestra de Adaptación de una Libreta de Uso Único

Percha de un agente Stasi con una *libreta de un sólo uso*
(extraído de: *Spiegel Spezial 1/1990)*

Menú:
”Cifrar/Descifrar” \
”Simétrico (clásico)” \
”Vernam”

Problema de Distribución de Claves

Distribución de claves para métodos de cifrado simétrico

Si **2 personas** se comunican utilizando un cifrado simétrico, **necesitan una clave secreta común.**

Si n personas se comunican entre ellas, entonces necesitan $\mathbf{S_n = n * (n-1) / 2}$ claves.

Esto significa que

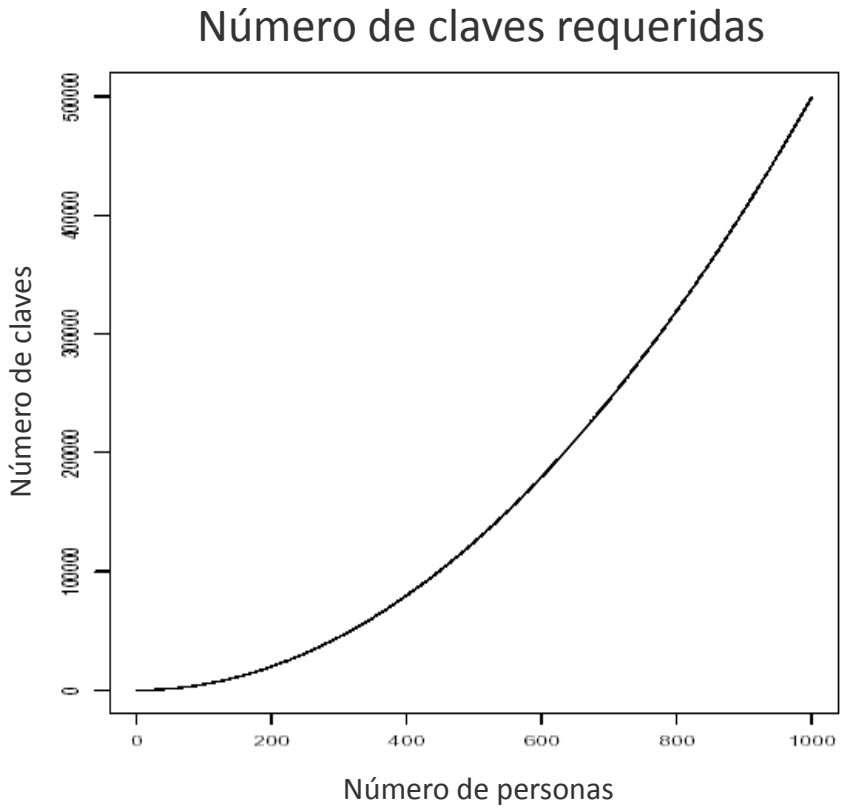
n = 100 personas requieren

$\mathbf{S_{100} = 4.950}$ claves; y

n = 1.000 personas requieren

$\mathbf{S_{1000} = 499.500}$ claves.

⇨ Un factor 10 de más personas, resulta un factor 100 de más claves



Criptografía – Conceptos Importantes (2)

Resolver el problema de distribución de clave mediante criptografía asimétrica

Criptografía Asimétrica

- Durante siglos se creía que: el emisor y el receptor necesitaban el mismo secreto.
- Ahora: Cada miembro necesita un par de claves (solución al problema de distribución de claves)

Cifrado Asimétrico

- „Todo el mundo puede cerrar un candado o puede dejar caer una carta en un buzón.“
- MIT, 1977: Leonard Adleman, Ron Rivest, Adi Shamir (más conocido como RSA)
- GCHQ Cheltenham, 1973: James Ellis, Clifford Cocks (aceptado públicamente en Diciembre de 1997)

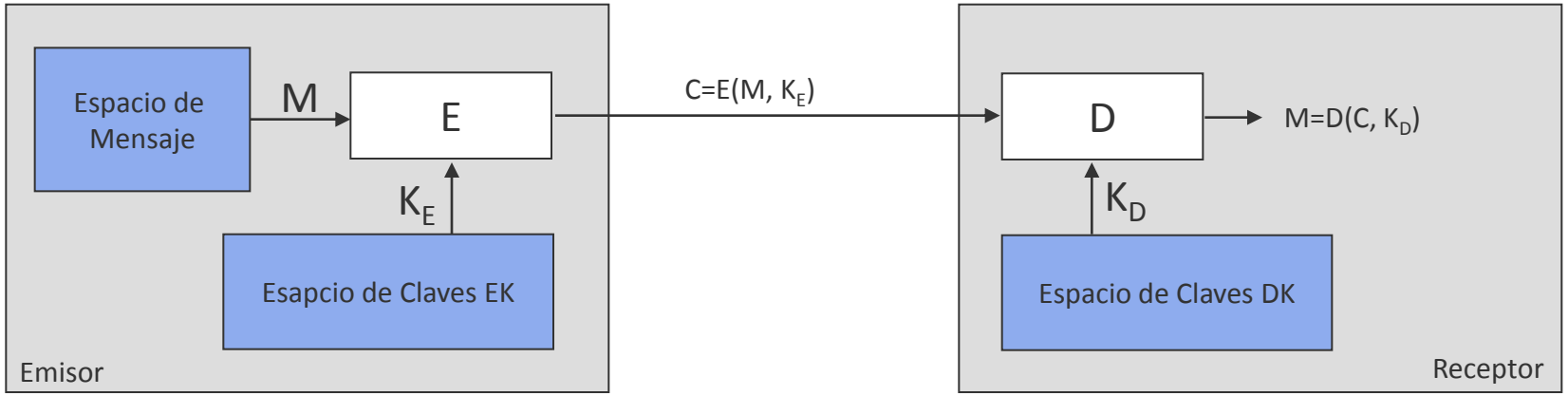
Distribución de claves

- Stanford, 1976: Whitfield Diffie, Martin Hellman, Ralph Merkle (Intercambio de clave Diffie-Hellman)
- GCHQ Cheltenham, 1975: Malcolm Williamson

¡La seguridad en redes abiertas (como Internet) sería extremadamente cara y compleja sin una criptografía asimétrica!

Cifrado y Descifrado

Cifrado Simétrico y asimétrico



a) Cifrado Simétrico: $\mathbf{K_E = K_D}$ (p.ej. AES)

b) Cifrado Asimétrico: $\mathbf{K_E \neq K_D}$ (p.ej. RSA)

secreto

público

privado/secreto

Criptografía – Conceptos Importantes (3)

La creciente relevancia de las matemáticas y las tecnologías de la información

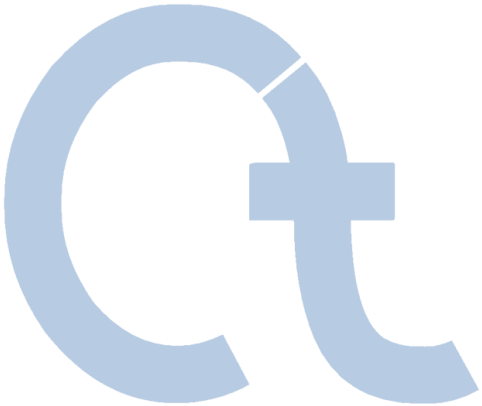
- **La criptografía moderna** se basa en las **matemáticas**
  - A pesar de los nuevos métodos de cifrado simétrico como el AES (mejor funcionamiento y una clave más corta comparados con los métodos asimétricos basados puramente en problemas matemáticos).
- La seguridad de los métodos de cifrado dependen fuertemente del estado en el que se encuentran las **matemáticas** y las **tecnologías de la información** (TI)
  - Complejidad computacional (el principal esfuerzo de procesado está relacionado con la longitud de la clave, demanda de dispositivos y complejidad de los datos)
    -> ver RSA: Bernstein, dispositivo TWIRL, RSA-160, RSA-200
  - Actividad muy alta en la investigación actual en:

    Factorización, algoritmos no paralelizables (a causa de los ordenadores cuánticos), mejor comprensión de la debilidad de los protocolos y los generadores aleatorios, ...).
- Grave Error: “Las matemáticas reales no tienen efecto sobre la guerra.” (G.H. Hardy, 1940)
- Los vendedores han descubierto la **seguridad** como un criterio esencial de **compra**.

Demostración con CrypTool

- *Análisis Estadístico*
- *Cifrar dos veces no siempre es mejor:*
  - *César: C + D = G (3 + 4 = 7)*
  - *Vigenère: - CAT + DOG = FOZ [(2,0,19)+(3,14,6)=(5,14,25)]*
    No hay mejoras, sin embargo usando:
    - GATO + PERRO = VEKFUPXFXOISXRHDKRKC)
    Se produce una clave mucho más fuerte.
- *Vernam (OTP)*
- *AES (clave de salida, análisis por fuerza bruta)*

Contenido



Características de CrypTool

eLearning

1. ¿Qué es CrypTool?

- Programa libre con interfaz gráfica
- Se pueden aplicar métodos criptográficos *y* analizarlos
- Completa ayuda en línea (comprensible sin un conocimiento profundo sobre criptografía)
- Contiene casi todas las funciones criptográficas actuales
- Introducción fácil tanto a la criptografía clásica como a la moderna
- No es una *"herramienta de hackers"*

2. ¿Por qué CrypTool?

- Origen en una iniciativa de concienciación de un instituto financiero
- Desarrollado en una cercana cooperación con universidades.
- Mejora en la educación universitaria y capacitación empresarial

3. Público Objetivo

- Grupo Principal: Estudiantes de informática, negocio informático y matemáticas
- Pero también para: usuarios de ordenador, desarrolladores de aplicaciones, empleados
- Prerrequisitos: conocimiento sobre PC
- Preferiblemente: Interesados en matemáticas y/o programación

Contenido del Paquete del Programa

alemán, inglés, polaco, castellano y serbio

Programa CrypTool

- Todas las funciones integradas en un *único* programa con una interfaz gráfica consistente
- Funciona sobre Win32
- Librerías Criptográficas de Secude y OpenSSL
- Aritmética de enteros grandes a través de Miracl, APFLOAT y GMP/MPIR, Reducción de base de retículos por NTL (V. Shoup)

Herramienta AES

- Programa independiente para cifrado AES (y creación de archivos autoextraíbles)

Juego Educativo

- „Number Shark“ estimula la comprensión de los factores y los números primos.

Completa Ayuda online (Ayuda HTML)

- Ayuda sensible al contexto disponible con F1 para todas las funciones del programa (incluidos los menús)
- Casos detallados de uso para muchas funciones del programa (tutorial)

Script (archivo .pdf) con información básica

- Métodos de cifrado • Factorización en Primos • Firma Digital
- Curvas Elípticas • certificado de clave pública • Teoría de Números Básica • Crypto 2020

Dos historias cortas relacionadas con la criptografía de Dr. C. Elsner

- „The Dialogue of the Sisters“ (una variante de RSA como elemento clave)
- „The Chinese Labyrinth“ (Tareas de teoría de números para Marco Polo)

Herramienta para el aprendizaje de Teoría de Números

Características (1)

Criptografía

Criptografía Clásica

- César (y ROT-13)
- Sustitución Monoalfabética (y Atbash)
- Vigenère
- Hill
- Sustitución Homofónica
- Playfair
- ADFGVX
- Suma de Bytes
- XOR
- Vernam
- Permutación / Trasposición (Rail Fence, Escítala, …)
- Solitario

Varias opciones para entender fácilmente los métodos criptográficos

- Alfabeto seleccionable
- Opciones: manejo de espacios, etc.

Criptoanálisis

Ataque a métodos clásicos

- Sólo texto cifrado
  - César
  - Vigenère (según Friedman + Schroedel)
  - Suma
  - XOR
  - Sustitución
  - Playfair
- Texto Claro conocido
  - Hill
  - Transposición de Columna Simple
- Manual (soportado)
  - Sustitución mono-alfabética
  - Playfair, ADFGVX, Solitario

Métodos de Análisis soportados

- Entropía, frecuencia real
- Histograma, análisis de n-grama
- Autocorrelación
- Periodicidad
- Análisis de aleatoriedad
- Base64 / UU-Encode

Características (2)

Criptografía

Cifrado simétrico moderno

- IDEA, RC2, RC4, RC6, DES, 3DES, DESX
- Candidatos AES de la última ronda de selección (Serpent, Twofish, …)
- AES (=Rijndael)
- DESL, DESXL

Cifrado Asimétrico

- RSA con certificados X.509
- Demostración RSA
  - Comprensión de ejemplos
  - Alfabeto y longitud de bloque seleccionable

Cifrado Híbrido (RSA + AES)

- Diagrama de flujo de datos interactivo

Criptoanálisis

Ataque por fuerza bruta para algoritmos simétricos

- Para todos los algoritmos
- Suposiciones:
  - La entropía de un texto claro es pequeña o la clave se conoce parcialmente o se conoce el alfabeto del texto claro

Ataque al cifrado RSA

- Factorización del módulo RSA
- Ataques de bases de Retículos

Ataque al cifrado híbrido

- Ataque a RSA o
- Ataque a AES (ataque del canal lateral)

Características (3)

Criptografía

Firma Digital

- RSA con certificados X.509
  - Firma como un diagrama de flujo de datos
- DSA con certificados X.509
- Curva Elíptica DSA, Nyberg-Rueppel

Funciones Hash

- MD2, MD4, MD5
- SHA, SHA-1, SHA-2, RIPEMD-160

Generadores Aleatorios

- Secude
- x^2 mod n
- Generador de congruencias Lineal (LCG)
- Generador de congruencias Inverso (ICG)

Criptoanálisis

Ataque a la firma RSA

- Factorización del módulo RSA
- Factible hasta los 250 bits o 75 decimales (en un PC estándar)

Ataque a las funciones hash / firma digital

- Generar colisiones hash para un texto en ASCII (paradoja del cumpleaños) (hasta 40 bit en unos 5 min)

Análisis de datos aleatorios

- Batería de pruebas FIPS-PUB-140-1
- Periodicidad, Vitany, entropía
- Frecuencia real, histograma
- Análisis de n-gramas, autocorrelación
- Test de compresión ZIP

Características (4)

Animaciones / Demostraciones

- César, Vigenère, Nihilist, DES (todo con ANIMAL)
- Enigma (Flash)
- Rijdael/AES (Flash)
- Cifrado y descifrado Híbrido (AES-RSA y AES-ECC)
- Generación y verificación de firmas digitales
- Intercambio de claves Diffie-Hellman
- Secreto compartido (con CRT o Shamir)
- Método Desafío-Respuesta (autentificación)
- Ataque del canal lateral
- E-mail seguro con protocolo S/MIME (con Java y Flash)
- Presentación gráfica en 3D de chorros de datos (aleatorios)
- Sensibilidad de funciones hash con respecto a cambios en el texto claro
- Teoría de Números y criptosistema RSA (con Authorware)

Características (5)

Funciones Adicionales

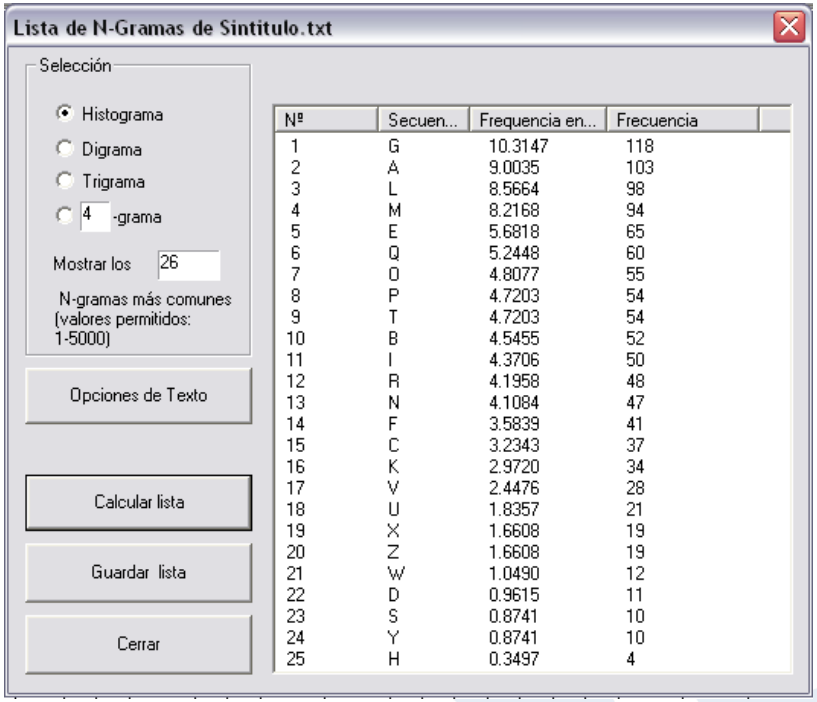
- Diferentes funciones para RSA y números primos
- Cifrado Homofónico y por permutación (Transposición Doble Columna)
- PKCS #12 importado y exportado para PSEs (Entorno Personal de Seguridad)
- Generar archivos has de archivos grandes sin cargarlos
- Ataques por fuerza bruta flexibles sobre cualquier algoritmo simétrico moderno.
- Demostración de ECC (como aplicación Java)
- Medidor de Calidad de Contraseñas (PQM)
- Múltiples opciones de texto para los cifrados clásicos (ver Ejemplo 24)
- Y mucho más …

Análisis de la Estructura de un Idioma

Opciones de análisis disponibles en CrypTool

Número de caracteres, n-grama, entropía

- ver menú “Análisis” \ “Herramientas para el Análisis” \ ...



Demonstración de Interactividad (1)

Análisis Vigenère

Demostración en CrypTool

El resultado del análisis de Vigenère puede rehacerse manualmente (cambiando la longitud de la clave):

1. Cifrar el ejemplo inicial con: **TESTETE**
   - *"Cifrar/Descifrar"* \ *"Simétrico (clásico)"* \ *"Vigenère"*
   - Introducir TESTETE ⇨ *"Cifrar"*

Análisis del resultado del cifrado:

- *"Análisis"* \ *"Cifrado Simétrico (clásico)"* \ *"Sólo texto cifrado"* \ *"Vigenère"*
- Longitud de clave deducida: 7, Clave deducida: TESTETE

2. Cifrar el ejemplo inicial con: **TEST**
   - *"Cifrar/Descifrar"* \ *"Simétrico (clásico)"* \ *"Vigenère"*
   - Introducir TEST ⇨ *"Cifrar"*

Análisis del resultado del cifrado:

- *"Análisis"* \ *"Cifrado Simétrico (clásico)"* \ *"Sólo texto cifrado"* \ *"Vigenère"*
- Longitud de clave deducida: 8 – Falso
- Longitud de clave seleccionada automáticamente a 4 (puede ajustarse manualmente)
- Clave deducida: TEST

Demonstración de Interactividad (2)

Factorización automatizada

Demostración en CrypTool

Factorización de un número compuesto con algoritmos de factorización

- Algunos métodos se ejecutan en paralelo (multihilo)
- Los métodos tienen ventajas e inconvenientes específicos (p.ej. Algunos métodos sólo pueden determinar factores pequeños)

Ejemplo de Factorización 1:

316775895367314538931177095642205088158145887517

Número decimal de 48-dígitos

=

3 * 1129 * 6353 * 1159777 * 22383173213963 * 567102977853788110597

Ejemplo de Factorización 2:

2^250 - 1

Número decimal de 75-dígitos

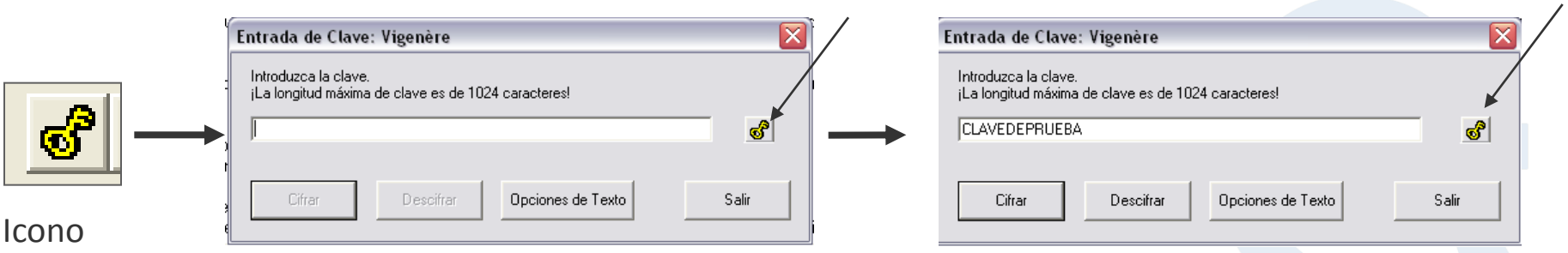
=

3 * 11 * 31 * 251 * 601 * 1801 * 4051 * 229668251 * 269089806001 * 4710883168879506001 * 5519485418336288303251

Menú: “Procedimientos Indiv.” \ “Criptosistema RSA” \ “Factorización de un Número”

Conceptos para una Interfaz de fácil manejo

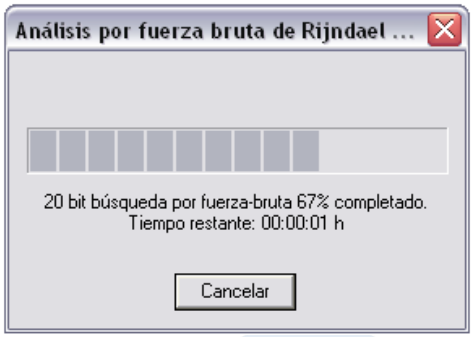
1. **Ayuda sensible al contexto (F1)**
   - F1 sobre una entrada de menú seleccionada nos muestra información sobre el algoritmo/método.
   - F1 en una ventana de diálogo explica la utilidad de la ventana.
   - Estas asistencias y los contenidos de los menús principales están vinculados de forma cruzada en la ayuda en línea.
2. **Pegar claves en una ventana de entrada de claves**
   - Se puede utilizar CTRL-V para pegar contenidos desde el porta papeles.
   - Las claves utilizadas se pueden obtener de una ventana de texto cifrado por medio de un icono de la barra de herramientas. Su correspondiente icono en la ventana de entrada de clave se puede utilizar para pegar la clave en el campo de entrada. Se utiliza un **depósito de claves interno** de CrypTool que está disponible para cada método (útil para claves largas y/o “específicas”- p.ej. en el cifrado homofónico).



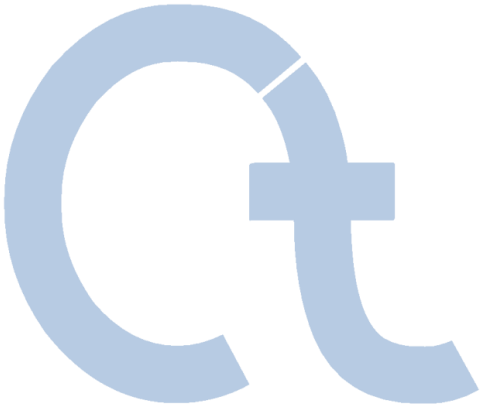
Icono

Desafíos para los Desarrolladores (Ejemplos)

1. **Muchas funciones trabajan en paralelo**
   - La factorización trabaja con algoritmos multihilo
2. **Alto Rendimiento**
   - Localizar colisiones hash (paradoja del cumpleaños) o ejecutar análisis por fuerza bruta
3. **Considera límites de memoria**
   - Algoritmo de Floyd (mapeados para localizar colisiones hash) o con una factorización con criba cuadrática
4. **Medida del tiempo y estimaciones**
   - Muestra el tiempo restante durante la fuerza bruta
5. **Reusabilidad / Integración**
   - Aplicaciones para la generación de números primos
   - Criptosistema RSA (cambia la vista después de un ataque exitoso de un usuario de clave pública al propietario de clave privada)
6. **Automatizar parcialmente la consistencia de funciones, GUI y ayuda en línea** (incluyendo varios idiomas y los SOs de Windows: XP, Vista y 7)



Contenido



Ejemplos de CrypTool

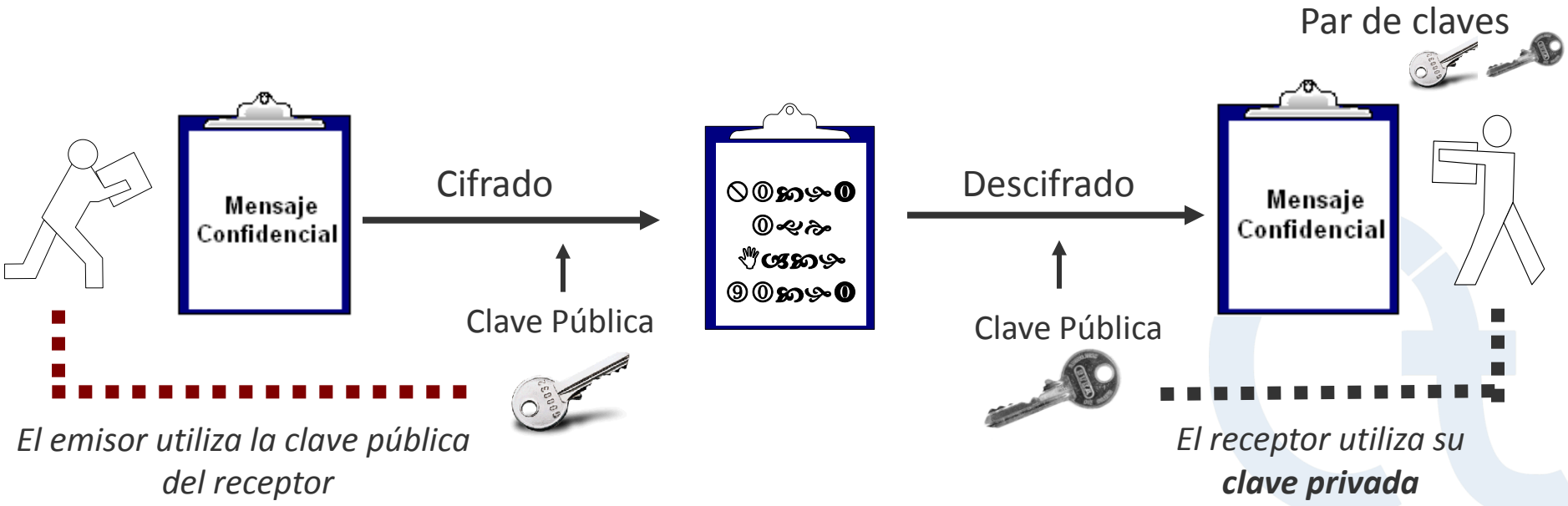
Visión general de los ejemplos

1. Cifrado con RSA / Test de primalidad / Cifrado híbrido y certificados digitales / SSL
2. Visualización de firma digital
3. Ataque al cifrado RSA (módulo N demasiado pequeño)
4. Análisis del cifrado en la PSION 5
5. Claves DES débiles
6. Localizando información de la clave (“clave NSA”)
7. Ataque a la firma digital por búsqueda de colisiones hash
8. Autentificación en un entorno cliente-servidor
9. Demostración de un ataque de canal lateral (en un protocolo de cifrado híbrido)
10. Ataque RSA utilizando reducción de retículos (lattice reduction)
11. Análisis de aleatoriedad con visualización 3-D
12. Secreto Compartido (Teorema Chino de los Restos (CRT) / Shamir)
13. Implementación del CRT en Astronomía
14. Visualización del cifrado utilizando ANIMAL
15. Visualización del AES
16. Visualización del cifrado Enigma
17. Visualización de E-mail seguro con S/MIME
18. Generación de un código de autentificación de un mensaje (HMAC)
19. Demo Hash
20. Herramienta de aprendizaje de teoría de números y cifrado asimétrico
21. Suma de puntos en curvas elípticas
22. Medidor de calidad de contraseñas
23. Análisis por Fuerza Bruta
24. Escítala / Rail Fence
25. Cifrado Hill / Análisis Hill
26. Ayuda online de CrypTool / Vista de árbol de menúes del programa

Ejemplos de CrypTool

Cifrado con RSA (en realidad, la mayoría de cifrados híbridos)

- **Bases para, por ejemplo, el protocolo SSL (acceso a sitios web protegidos)**
- **Cifrado asimétrico utilizando RSA**
  - Cada usuario tiene un par de claves– una pública y otra privada
  - El emisor cifra con la clave pública del receptor
  - El receptor descifra con su clave privada
- **Normalmente se implementa combinándolo con métodos simétricos (transferencia de la clave simétrica por codificación/descodificación RSA)**



Ejemplos (1)

Cifrado utilizando RSA – trasfondo Matemático / algoritmo

- Clave Pública: (n, e)
- Clave Privada: (d)

donde:

p, q grandes, números primos elegidos aleatoriamente con n = p*q;

d se calcula bajo las constantes mcd[φ(n),e] = 1; e*d ≡ 1 mod φ(n).

Operación de cifrado y descifrado: $(m^e)^d \equiv m \bmod n$

- n es el módulo, cuya longitud en bit se refiere a la longitud de la clave RSA.
- mcd = máximo común divisor.
- φ(n) es la función fi de Euler.

Procedimiento :

- Transformación del mensaje en representación binaria
- Mensaje Cifrado m = $m_1,...,m_k$ en el sentido de los bloques, con para todo m_j: $0 \leq m_j < n$; tamaño de bloque máximo r, por eso: $2^r \leq n$ $(2^r-1 < n)$

Vea también: Animación flash interactiva sobre los fundamentos del cifrado RSA:
http://cryptool.com/download/RSA/RSA-Flash-en/player.html

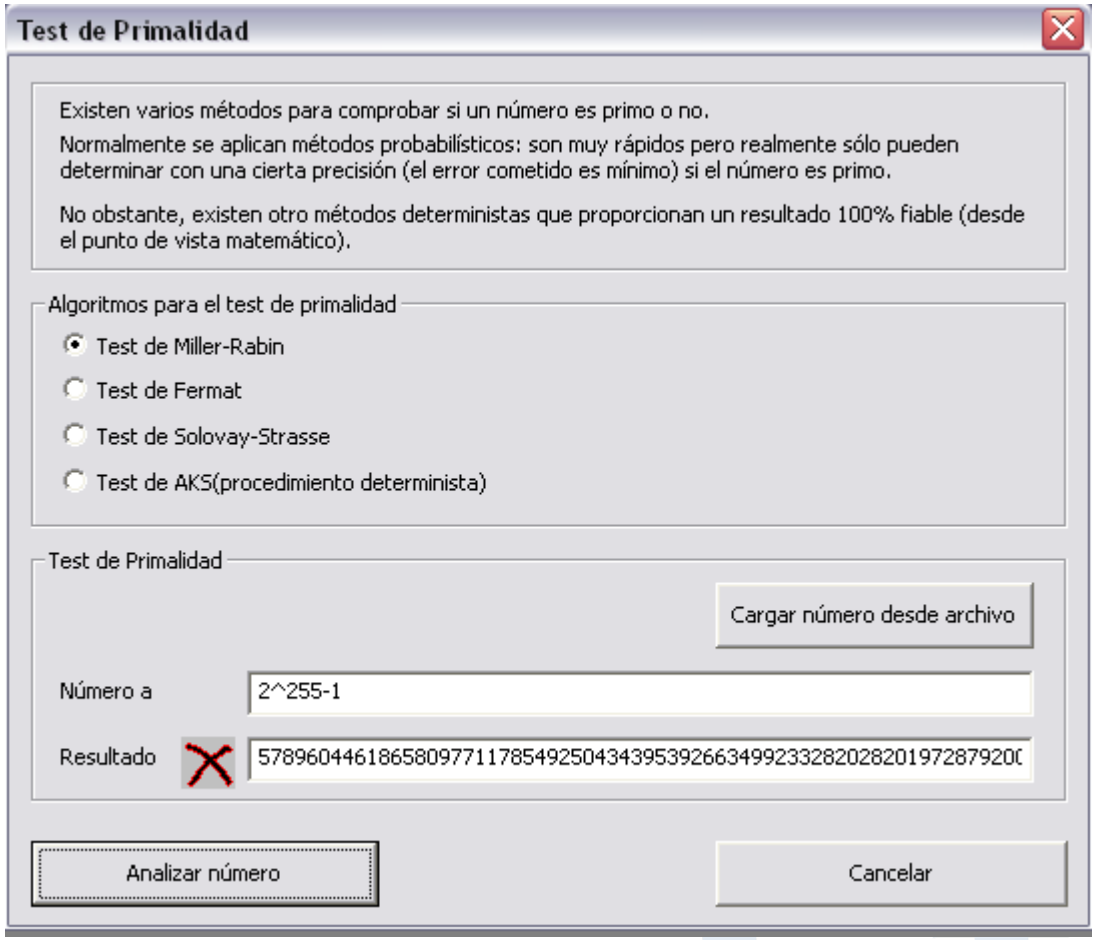
Ejemplos (1)

Test de Primalidad – Se necesita para los enormes primos de RSA.

- Pruebas probabilísticas rápidas
- Pruebas Deterministas

Los métodos de prueba de números primos se realizan mucho más rápido si un número grande es primo, entonces los métodos de factorización conocidos pueden separar un número de tamaño similar en sus factores primos.

Para los test AKS se integraron a CrypTool las bibliotecas GMP (**G**NU **M**ultiple **P**recision Arithmetic Library) y MPIR (**M**ultiple **P**recision **I**ntegers and **R**ationals).



Menú: "Procedimientos. Indiv" \ "RSA Criptosistema" \ "Test de Primalidad"

Observación: 2^255 - 1 = 7 * 31 * 103 * 151 * 2143 * 11119 * 106591 * 131071 * 949111 * 9520972806333758431 * 57024515776397755458386431 51

Ejemplos (1)

El mayor número primo descubierto hasta ahora – Números primos de Mersenne

Los números primos más grandes descubiertos, son los denominados: Números primos de Mersenne.

El número actualmente recordista tiene 12.978.189 dígitos decimales y fue descubierto en 2008 por el grupo del proyecto GIMPS.

En el siguiente diálogo de CrypTool, podrá calcular y mostrar rápidamente cada una de sus cifras en un archivo.

Para eso, la biblioteca APFLOAT fue integrada al programa.

En el menú contextual de cada uno de los campos de entrada o salida puede des/activar el separador de miles.

Nota: 2^43.112.609 - 1 = 316.470.269 … 697.152.511

Los números extremadamente grandes no deberían ser seleccionados y copiados directamente del campo "Resultado"; para no perjudicar la performance de la GUI. Por favor, utilice el botón "Escribir resultado" para poder mostrar el resultado en su totalidad en la misma ventana de CrypTool.

Menú: "Procedimientos Indiv." \ "Teoría de Números – Interactiva" \ "Calcular números de Mersenne"

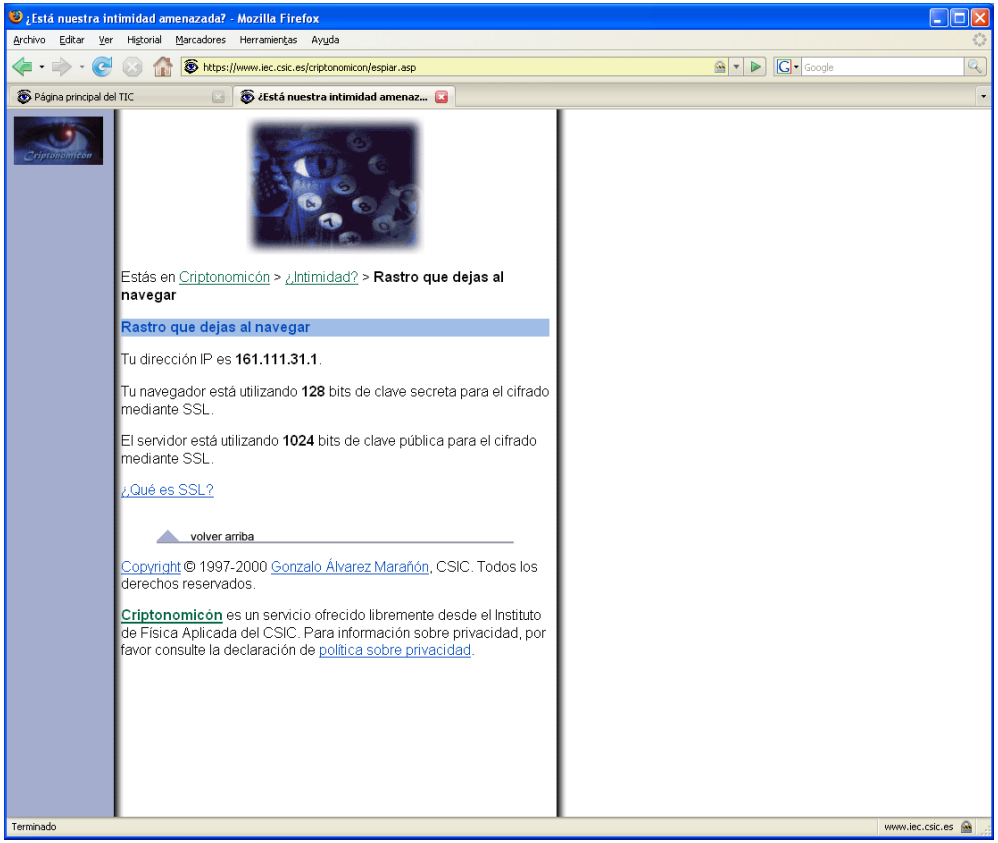
Ejemplos (1)

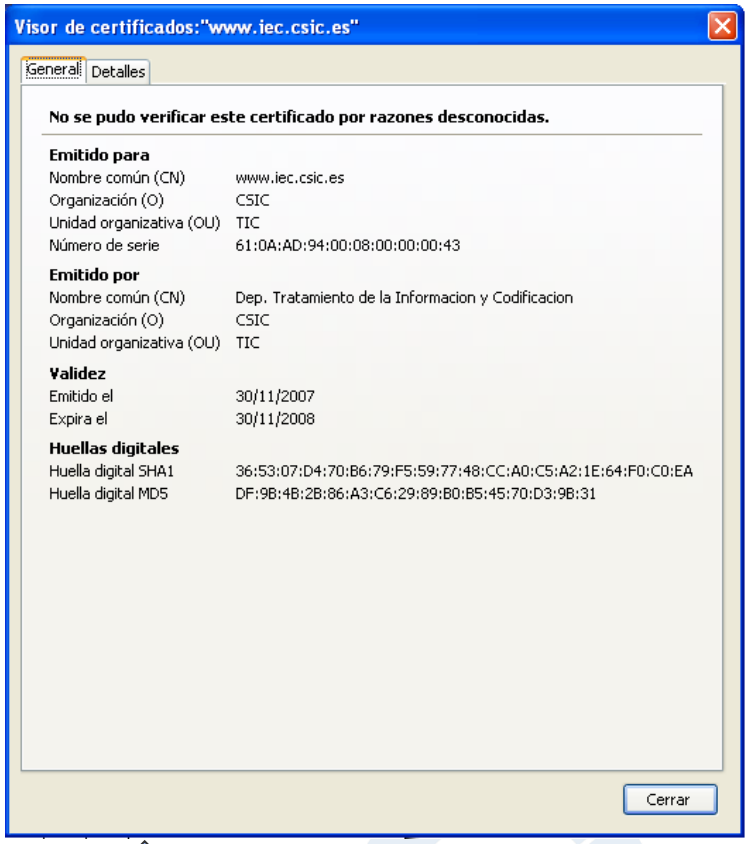
Cifrado Híbrido y certificados digitales

- Cifrado Híbrido – **Combinación de cifrado simétrico y asimétrico**
  1. Generación de una clave simétrica aleatoria (clave de sesión)
  2. Se transfiere la clave de sesión – protegida por una clave asimétrica
  3. Se transfiere el mensaje – protegida por la clave de sesión
- Problema: **Ataques del *hombre en el medio* – ¿La clave pública del receptor pertenece realmente al receptor?**
- Solución: Certificados Digitales – **Una central (p. ej. Telesec, VeriSign, Deutsche Bank PKI), en la que confían los usuarios, asegura la autenticidad del certificado y la clave pública contenida (parecido al pasaporte expedido por el estado).**
- El cifrado híbrido basado en certificados digitales **es la base para todas las comunicaciones electrónicas seguras:**
  - Compra por Internet y Banca Online
  - Correo electrónico seguro

Ejemplos (1)

Conexión online segura utilizando SSL y certificados

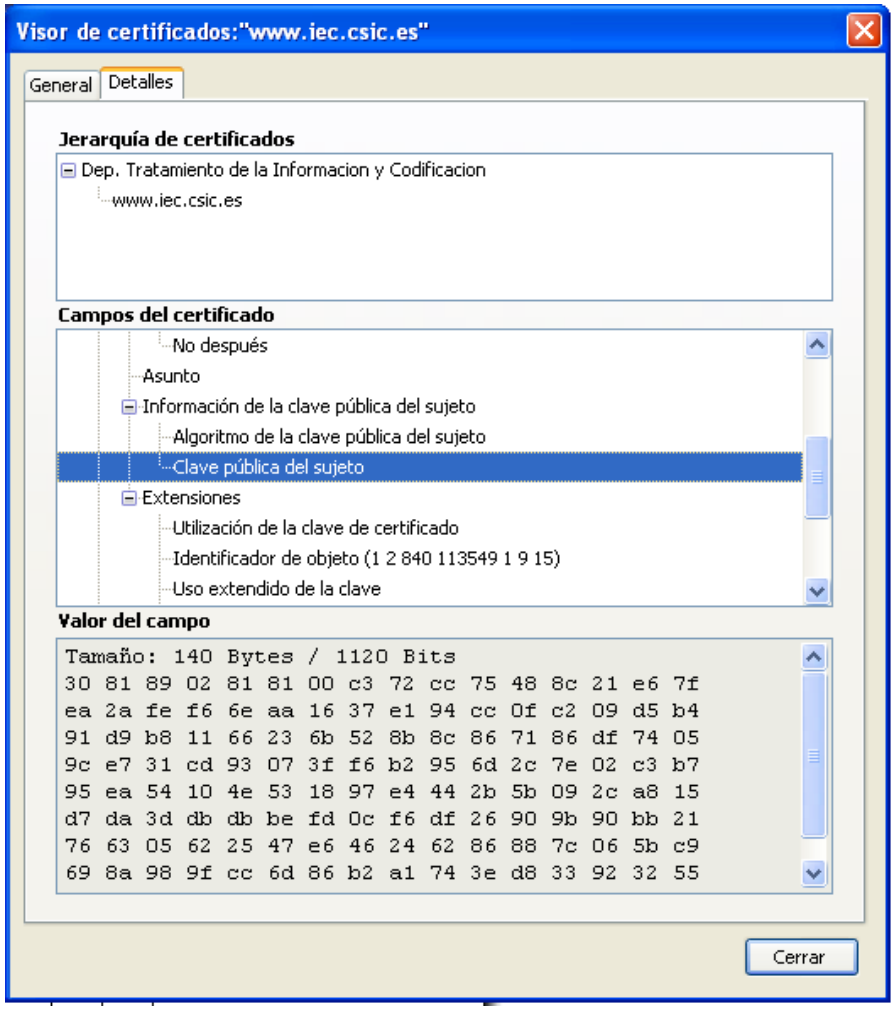




Esto significa que la conexión es autentificada y (al menos en un sentido) la transferencia de datos está fuertemente cifrada.

Ejemplos (1)

Atributos o campos de un certificado



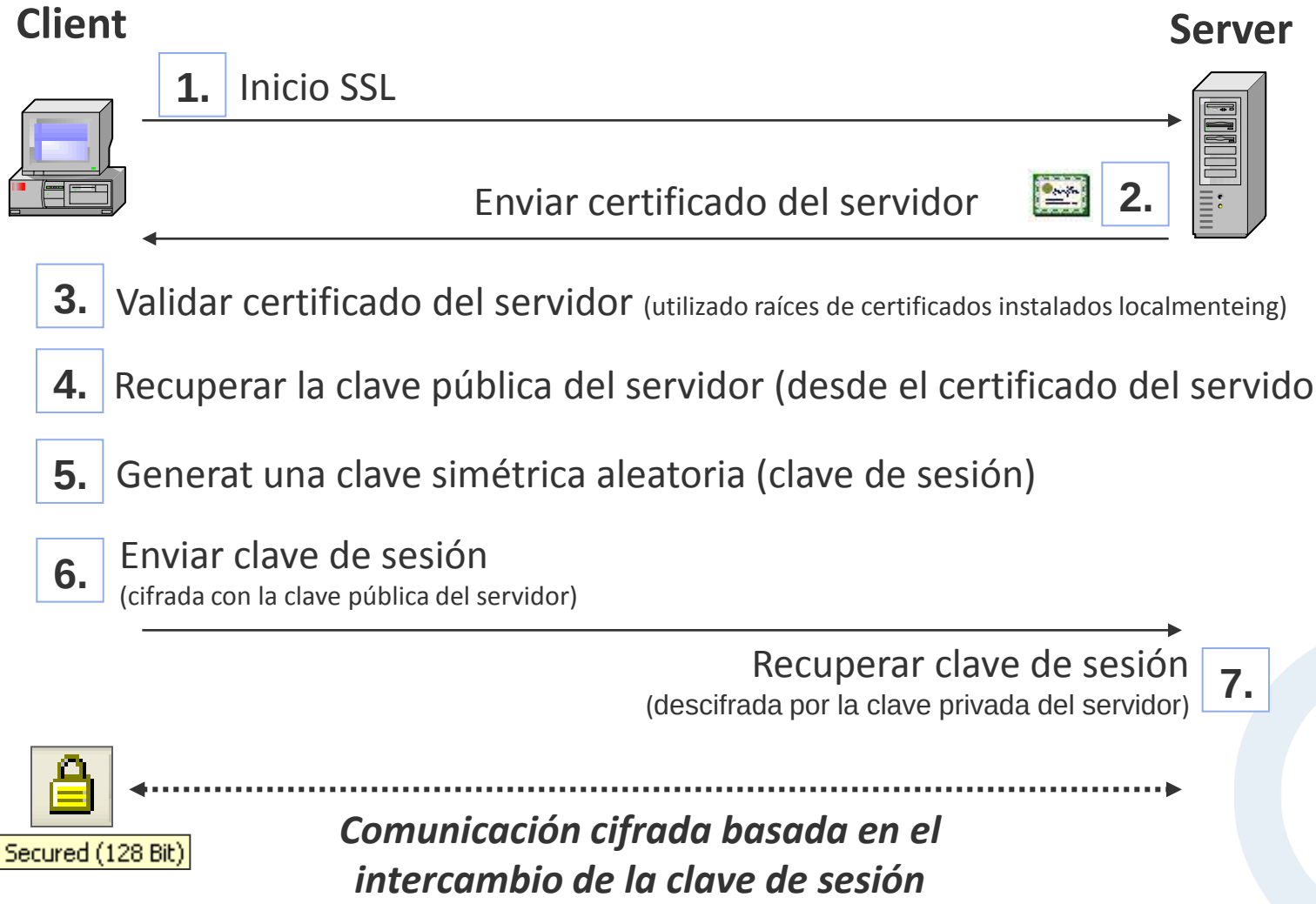
Atributos generales / campos

- Emisor (p.ej. VeriSign)
- Solicitante
- Período de validez
- Número de Serie
- Tipo de Certificado/ Versión (X.509v3)
- Algoritmo de firma
- Clave Pública (y método)

Clave Pública

Ejemplos (1)

Establecer una conexión segura SSL (Autenticación del Servidor)



Ejemplos (1)

Establecer una conexión segura SSL (Autenticación del Servidor)

General

- El ejemplo muestra el establecimiento de una conexión SSL típica para transferir datos delicados a través de internet (p.ej. compra online).
- Al establecer la conexión SSL solamente se autentifica el servidor utilizando el certificado digital (la autentificación del usuario se da normalmente a través del nombre de usuario y su contraseña después de que se haya establecido la conexión SSL).
- SSL también ofrece la opción de la autentificación del cliente basada en certificados digitales.

Comentarios al establecimiento de la conexión SSL (ver diapositiva anterior)

- Paso 1: Inicialización SSL – durante esta fase, se negocian las características tanto de la clave de sesión (p.ej. Tamaño en bits) como del algoritmo de cifrado simétrico (p.ej. 3DES, AES).
- Paso 2: En el caso de una jerarquía de certificado multinivel, también se pasan los certificados intermedios al cliente.
- Paso 3: En esta fase los certificados raíz instalados en la memoria de certificados del navegador se utilizan para validar el certificado del servidor.
- Paso 5: La clave de sesión se basan en las características negociadas (ver 1).

Ejemplos (2)

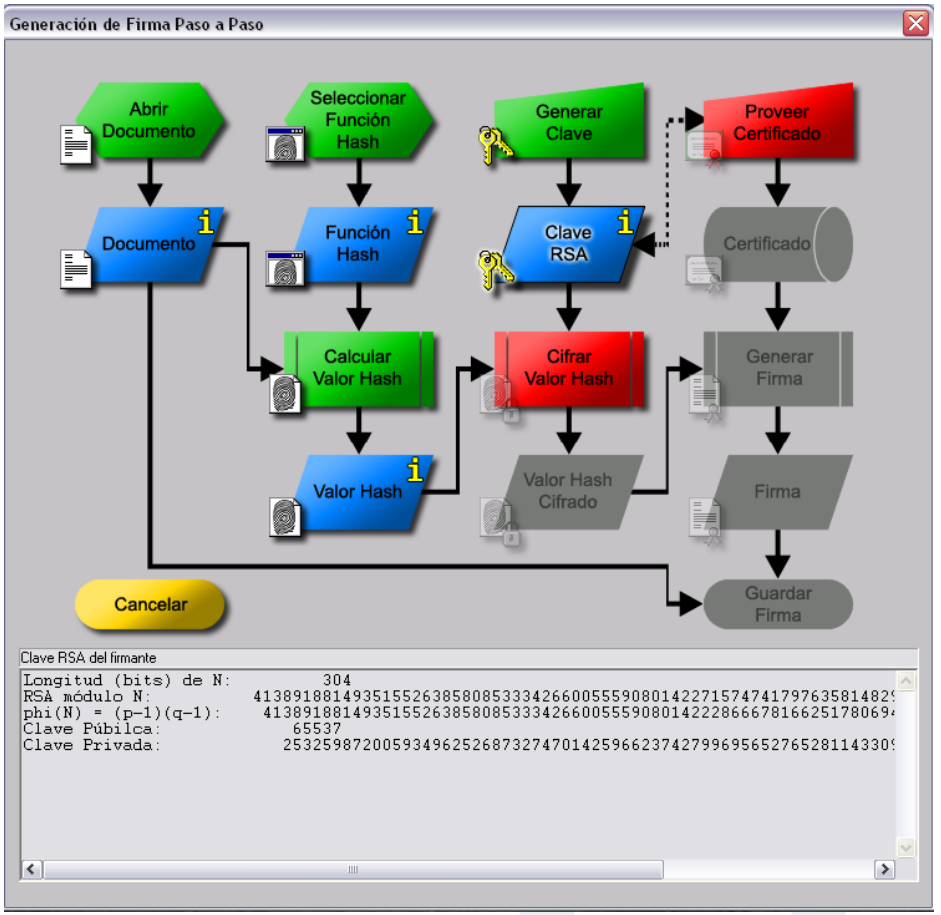
Visualización de una firma digital

Firma Digital

- Importancia creciente:
  - Equivalencia con la firma manual
    (ley de la firma digital)
  - Cada vez más utilizada en la industria,
  - Gobierno y usuarios
- Poca gente sabe cómo funciona exactamente

Visualización en CrypTool

- Diagrama de flujo de datos interactivo
- Parecida a la visualización del cifrado híbrido



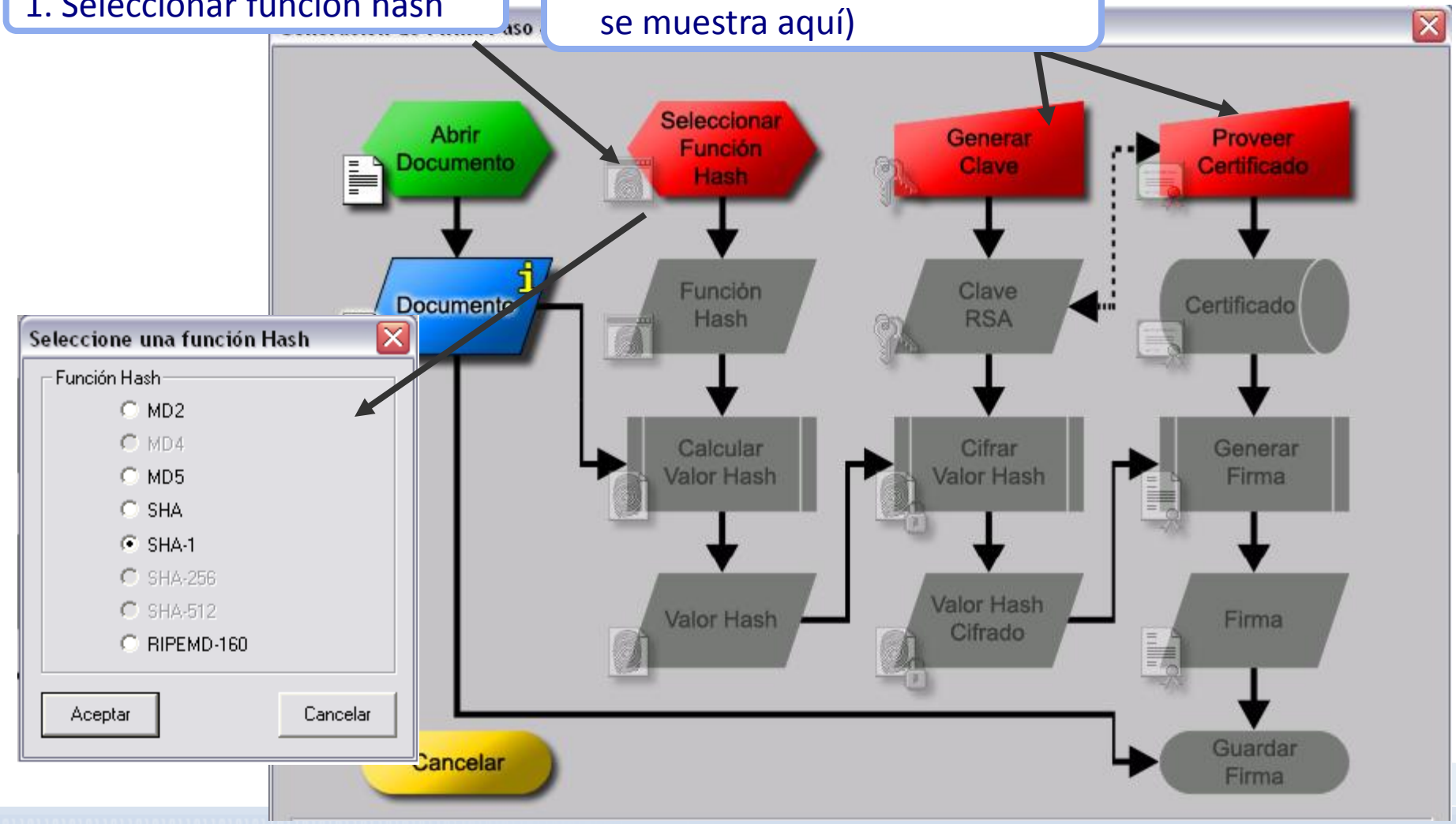
Menú: „Firma Digital/PKI“ \
„Demostración de firma(Generación de Firma)“

Ejemplos (2)

Visualización de una firma digital : a) Preparación

1. Seleccionar función hash

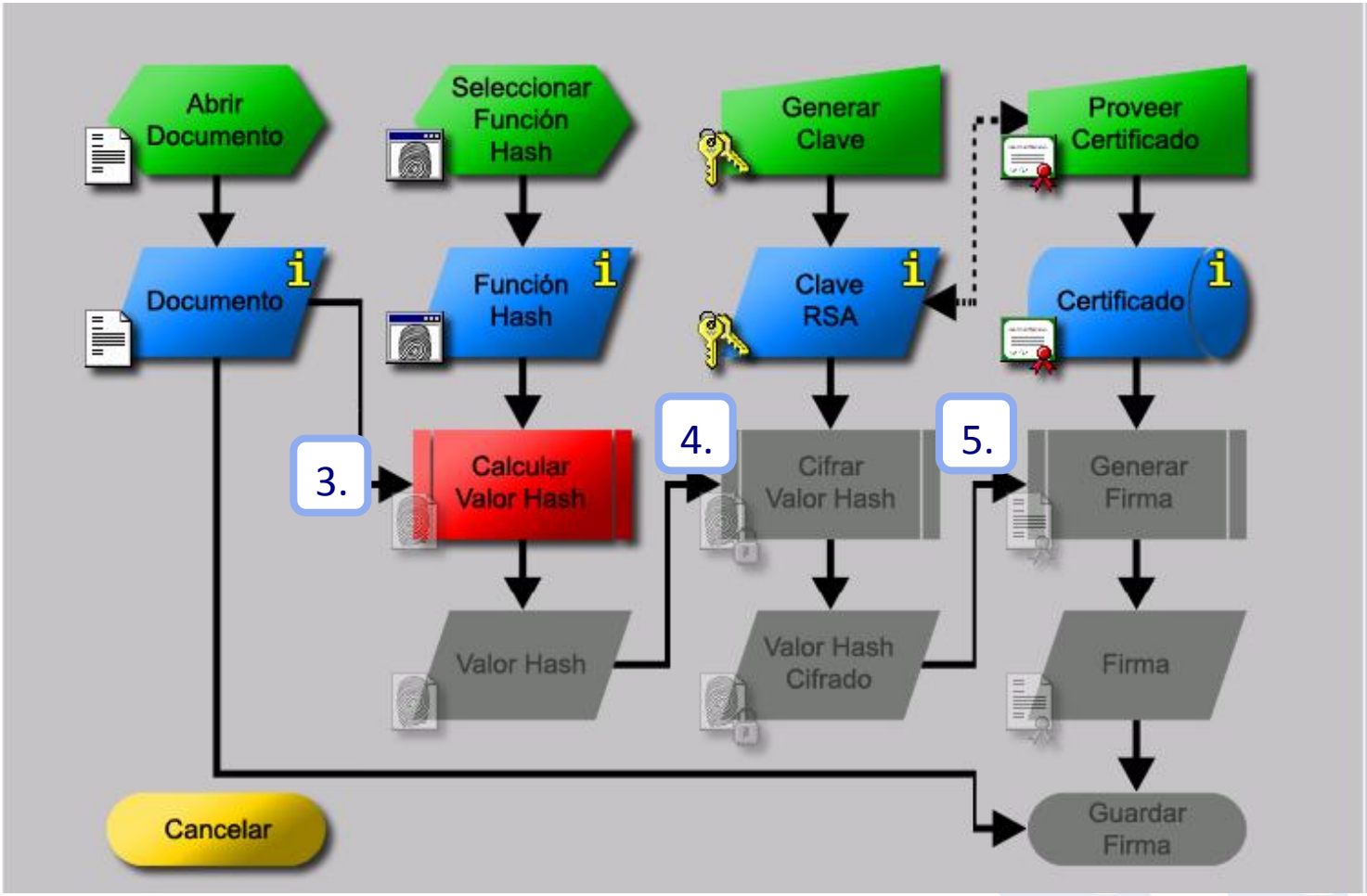
2. Facilitar clave y certificado (no se muestra aquí)



Ejemplos (2)

Visualización de una firma digital : b) Criptografía

3. Calcular valor hash
4. Cifrar el valor hash con la clave privada (firmar)
5. Generar firma

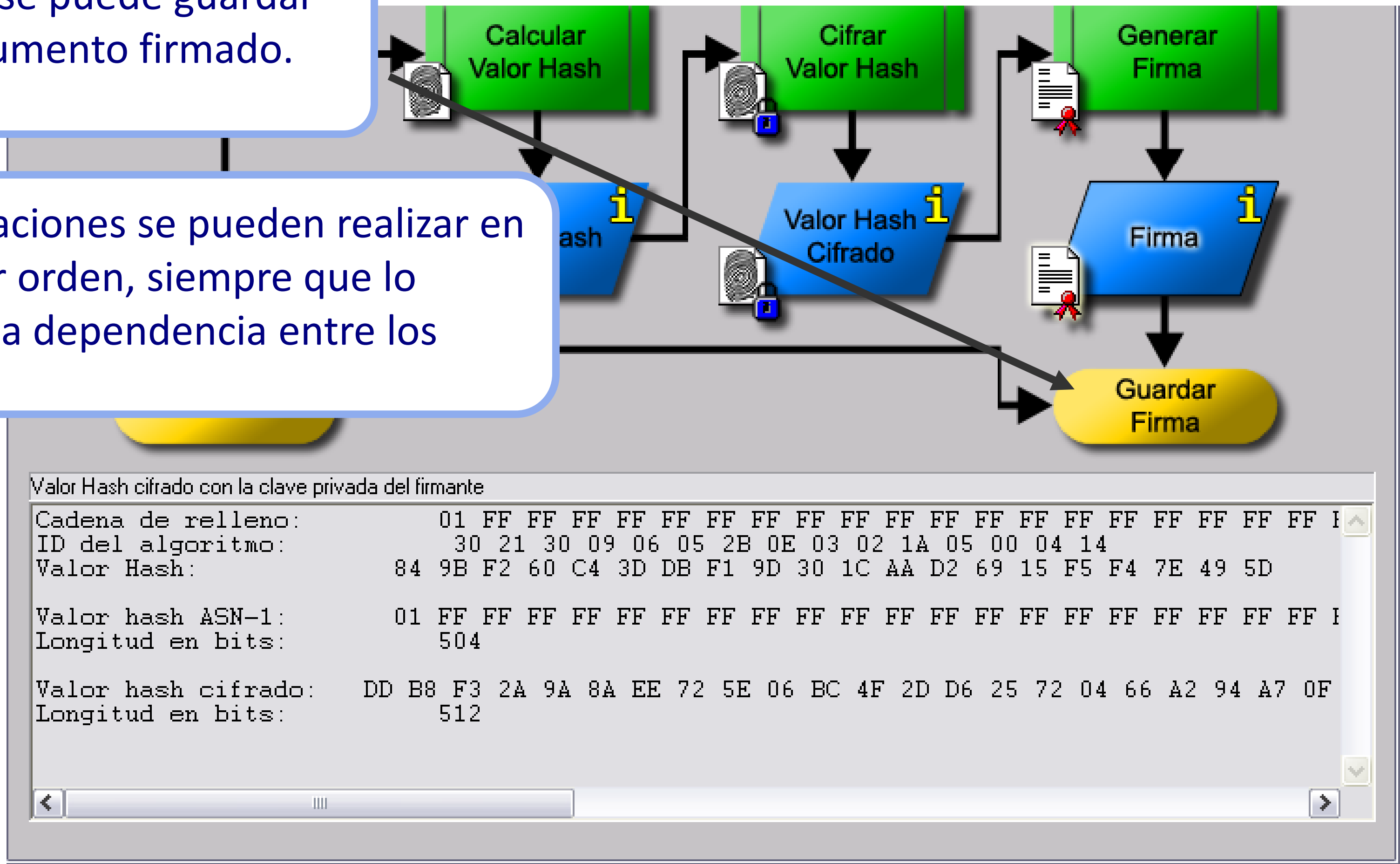


Ejemplos (2)

Visualización de una firma digital : c) Resultado

6. Ahora se puede guardar el documento firmado.

Las operaciones se pueden realizar en cualquier orden, siempre que lo permita la dependencia entre los datos.



Ejemplos (3)

Ataque al cifrado RSA con un RSA de módulo pequeño

Ejemplo de *Song Y. Yan*, Number Theory for Computing, Springer, 2000

- Clave pública
  - Módulo RSA **N = 639784868795271438588314150 41** (95 bit, 29 dígitos decimales)
  - exponente publico **e = 17579**
- Texto cifrado(longitud de bloque = 8):

  C_1 = 454116678950249382092592534 23,
  C_2 = 165970916214320200763115522 01,
  C_3 = 464689792797503547326376310 44,
  C_4 = 328701675459037413398196713 79

- ¡el texto debe ser descifrado!

¡El texto cifrado no se necesita para el criptoanálisis actual (localizar la clave privada)!

Solución utilizando CrypTool **(de forma más detallada en la sección de la ayuda online)**

- Introducir parámetros públicos en “RSA Criptosistema ” (menú: “Procedimientos Indiv. ”)
- Botón “Factorizar módulo RSA” produciendo los dos factores primos pq = N
- Basado en la información del exponente privado se determina $d=e^{-1} \bmod (p-1)(q-1)$
- Descifrar el texto cifrado con d: $M_i = C_i^d \bmod N$

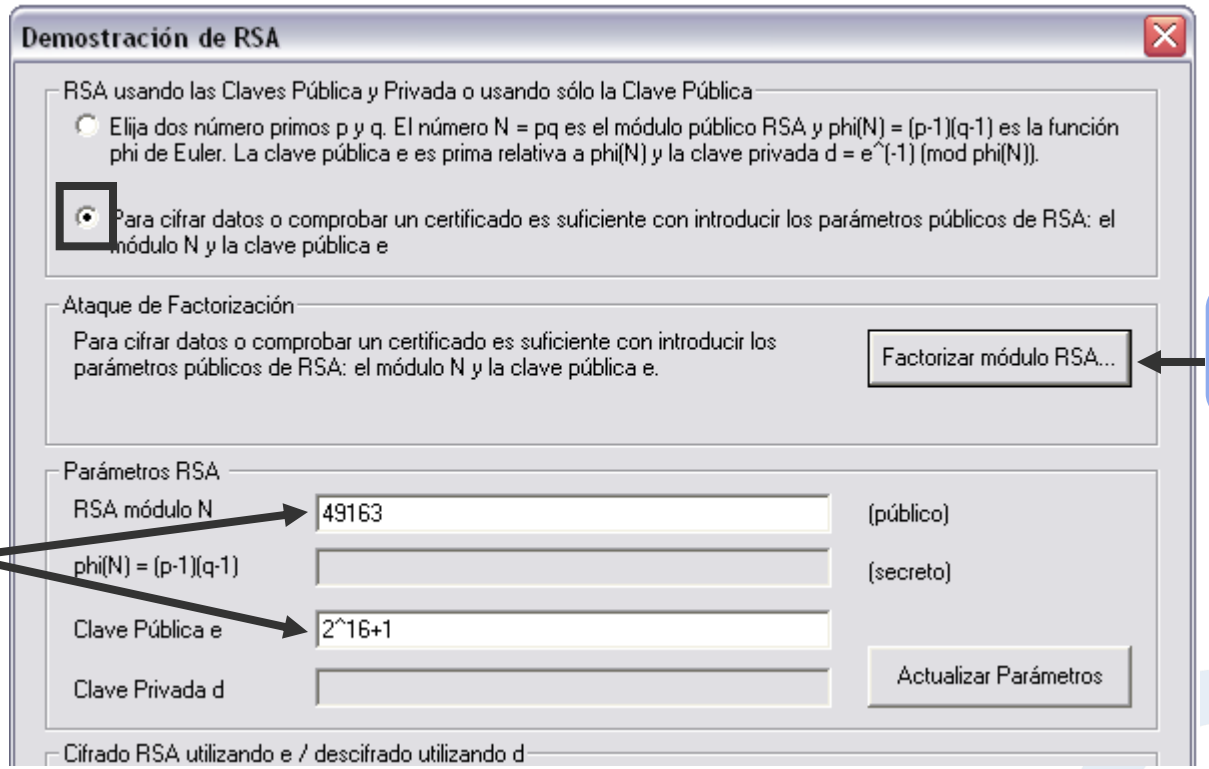
El ataque con CrypTool funciona para el módulo RSA hasta 250 bits.

¡Entonces podría firmar digitalmente por otra persona!

Ejemplos (3)

RSA de módulo pequeño: Introducir parámetros públicos del RSA

Menú: “Procedimientos. Indiv.” \ “RSA Criptosistema” \ “RSA Demonstración …”

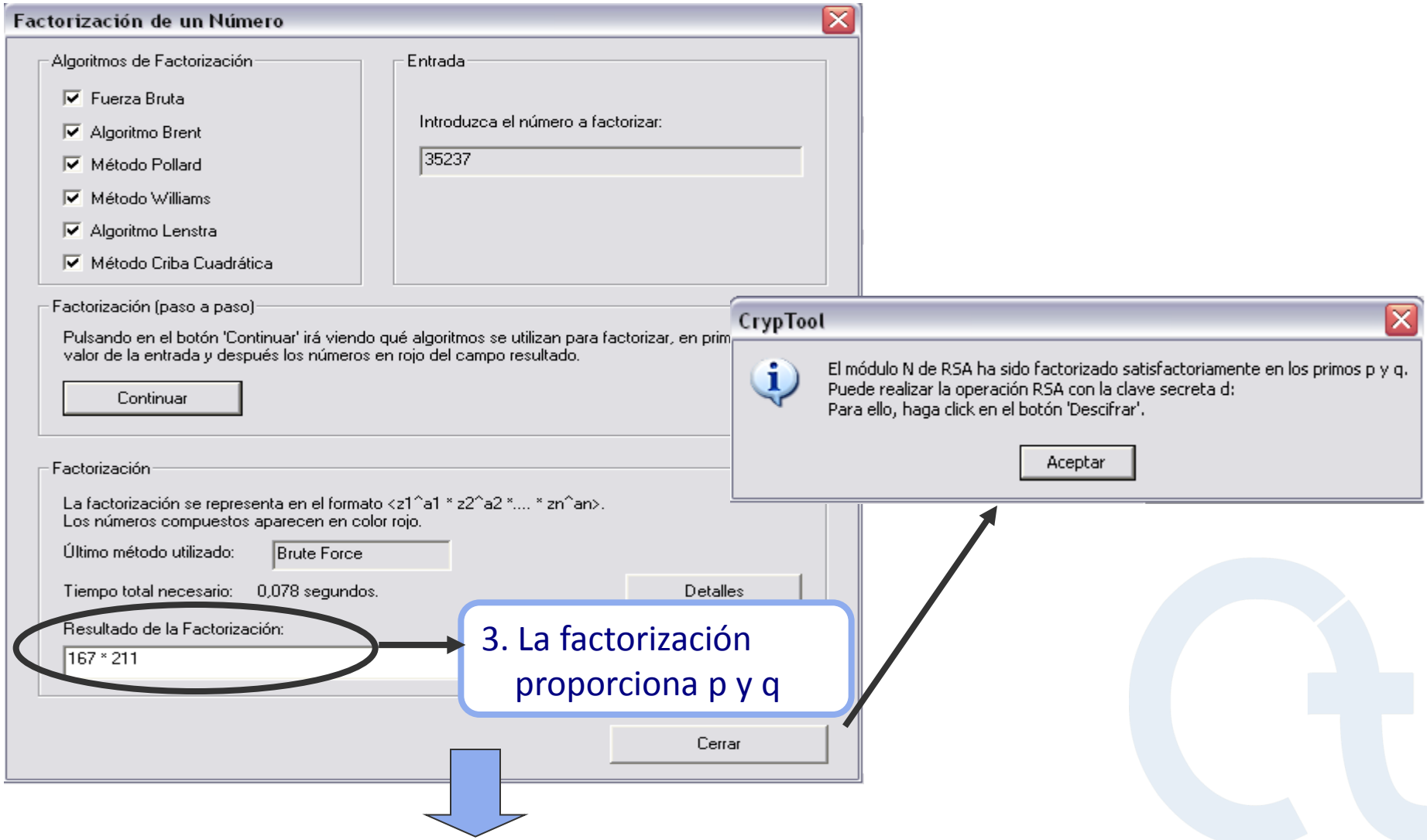


1. Introducir parámetros: “N” y “e”

2. Factorizar

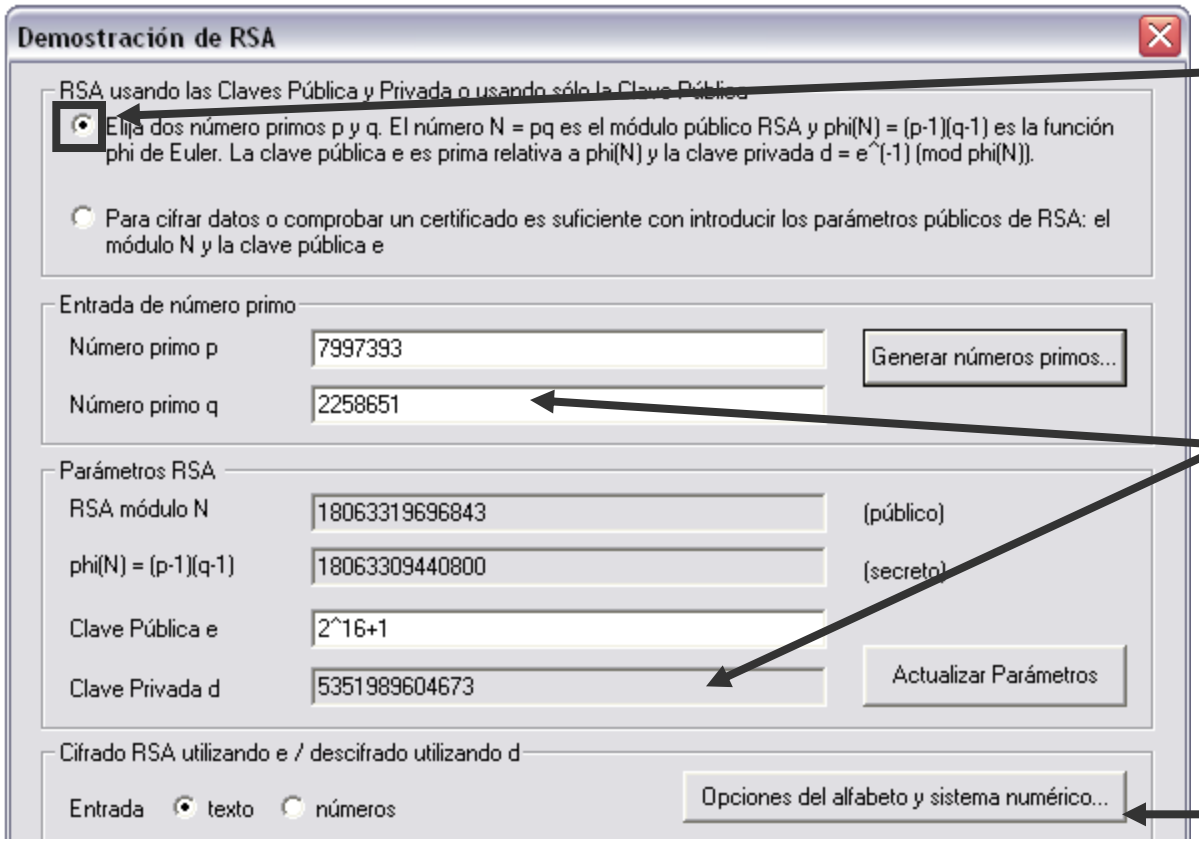
Ejemplos (3)

RSA de módulo pequeño: Factorizar módulo RSA



Ejemplos (3)

RSA de módulo pequeño: determinar la clave privada d



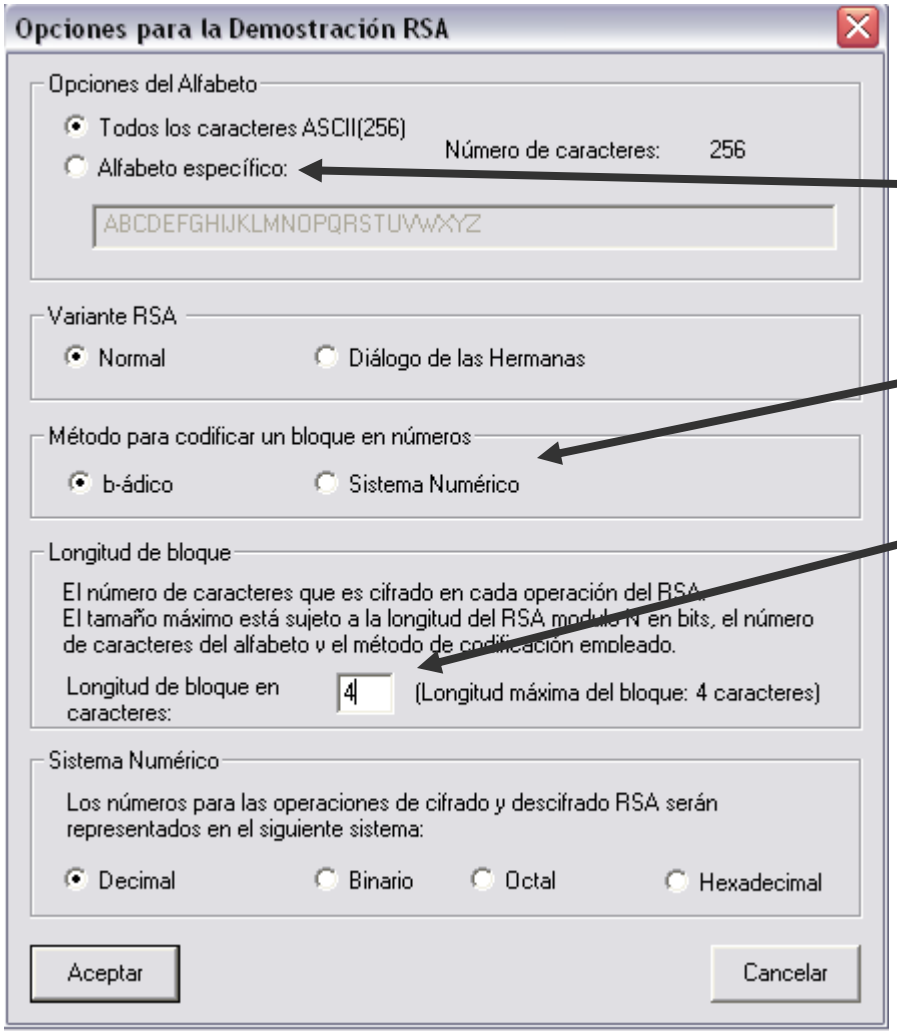
Cambia la visión del propietario de la clave secreta.

4. p y q se han introducido automáticamente y se ha calculado la clave privada

5. Ajustar Opciones

Ejemplos (3)

RSA de módulo pequeño: Ajustar Opciones



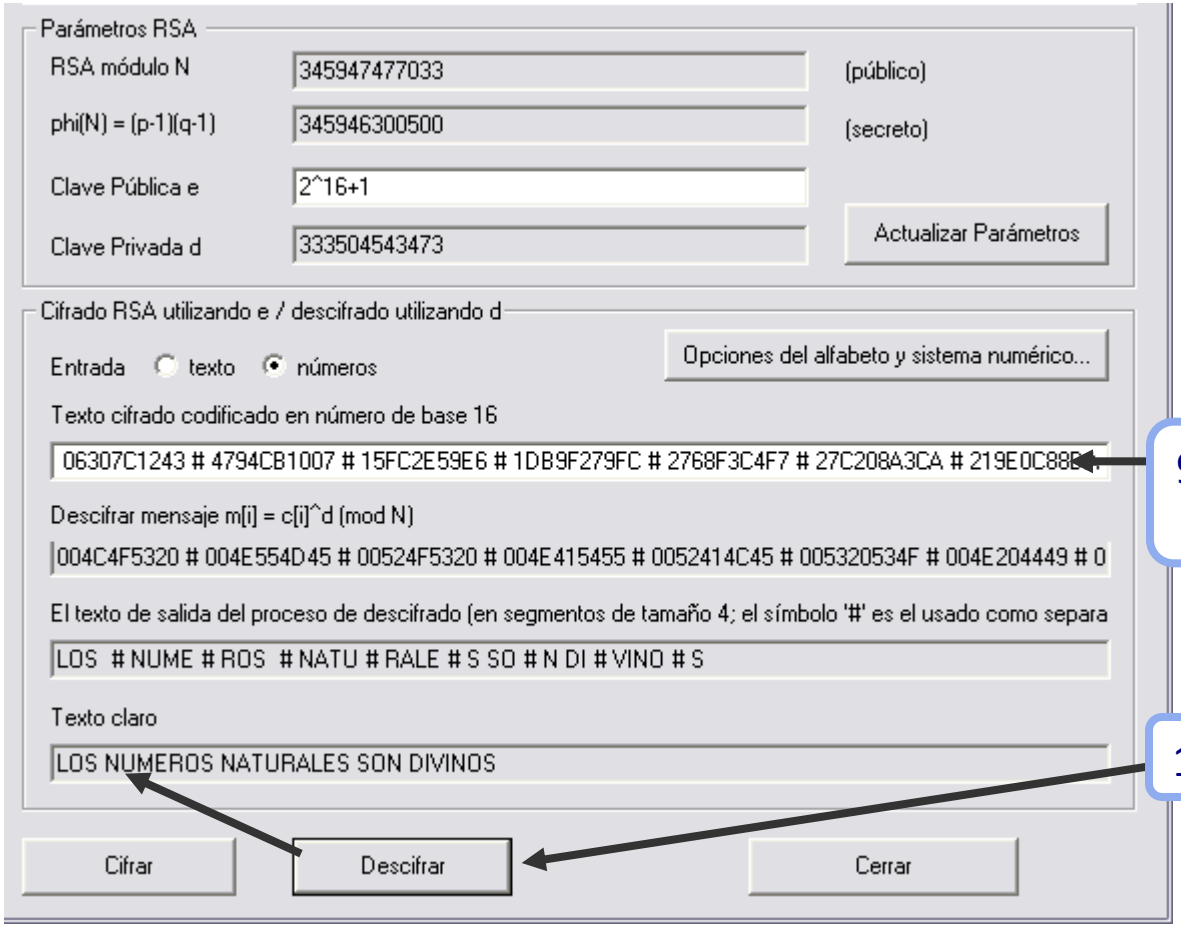
6. Seleccionar alfabeto

7. Seleccionar método de codificación

8. Seleccionar longitud de bloque

Ejemplos (3)

RSA de módulo pequeño: Descifrar texto cifrado



9. Introducir texto cifrado

10. Descifrar

Ejemplos (4)

Análisis del cifrado utilizado en la PSION 5

Aplicación práctica del criptoanálisis:

Ataque a la opción de cifrado de la aplicación de Procesador de texto de la PDA PSION 5

Punto de partida: un archivo cifrado con PSION

Requisitos

- Texto en alemán o en inglés cifrado
- Dependiendo del método y la longitud de la clave, texto desde 100 Bytes hasta varios kB

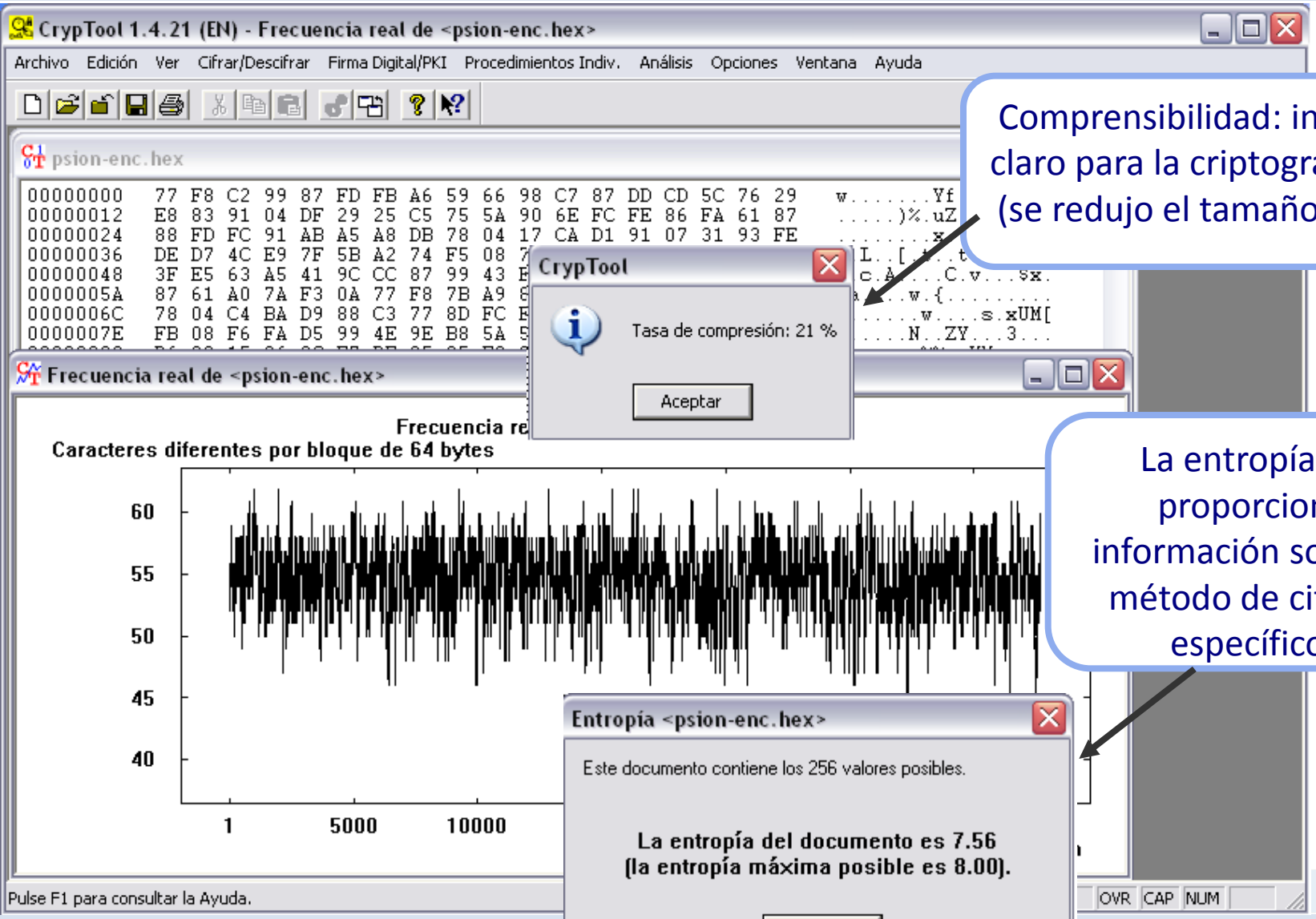
Procedimiento

- preanálisis
  - entropía
  - Entropía real
  - Test de compresión

  } *Probablemente un algoritmo de cifrado clásico*
- autocorrelación
- Intentar análisis automático con métodos clásicos

Ejemplos (4)

PDA PSION 5 – determinar entropía, test de compresión

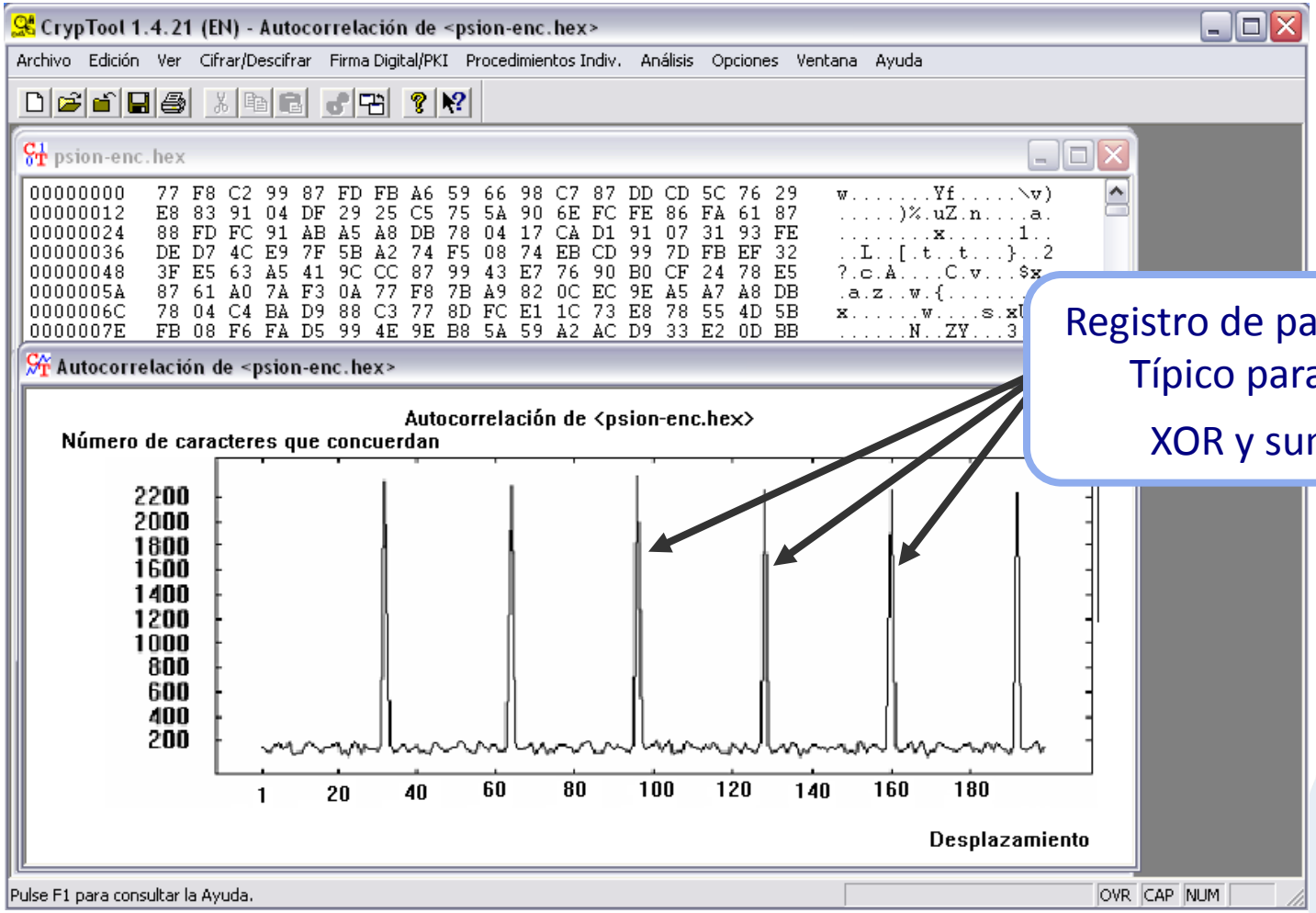


Comprensibilidad: indicador claro para la criptografía débil (se redujo el tamaño un 21%)

La entropía no proporciona información sobre el método de cifrado específico.

Ejemplos (4)

PDA PSION 5 – Determinar auto-correlación



Registro de patrón distintivo:
Típico para Vigenère,
XOR y suma binaria

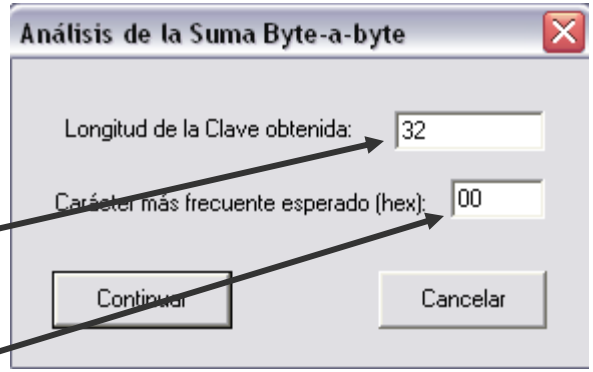
* El archivo cifrado está disponible en CrypTool (ver: CrypTool\examples\psion-en-enc.hex)

Ejemplos (4)

PDA PSION 5 – análisis automático

Análisis automático utilizando

- **Vigenère: sin éxito**
- **XOR: sin éxito**
- **Suma binaria**
  - CrypTool calcula la longitud de la clave utilizando la auto-correlación: 32 bytes
  - El usuario puede elegir el carácter que se espera que aparezca con mayor frecuencia:
    "e" = 0x65 (código ASCII)
  - El análisis calcula la clave más probable (basada en las suposiciones sobre la distribución)
  - Resultado: bueno, pero no perfecto

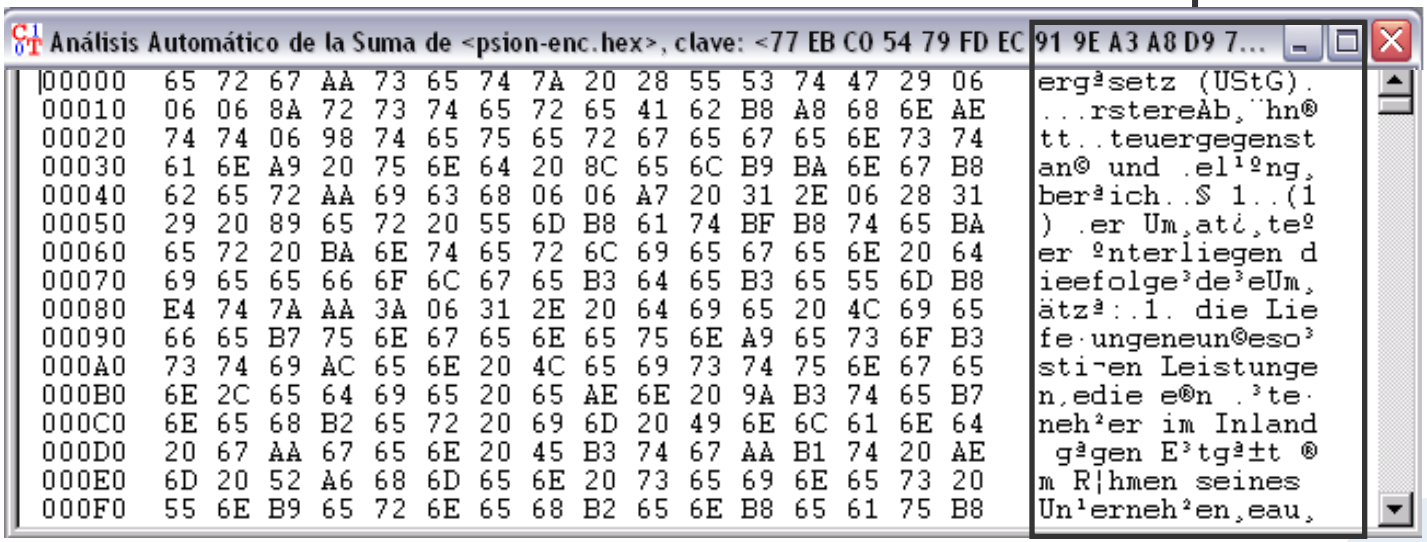


Ejemplos (4)

PDA PSION 5 – resultados del análisis automático

Resultados del análisis automático asumiendo “suma binaria”:

- El resultado es bueno, pero no perfecto: 24 de los 32 bytes de la clave son correctos.
- Se determina correctamente la longitud de la clave:32 .



- la contraseña introducida no tenía 32 bytes de longitud.
  ⇒ PSION Word deduce la clave actual de la contraseña.
- El post-procesado manual produce el texto cifrado (no se muestra)

Ejemplos (4)

PDA PSION 5 – determinar los bytes de clave restantes

Copiar la clave en el porta papeles durante el análisis automático

En análisis automático en hex dump,

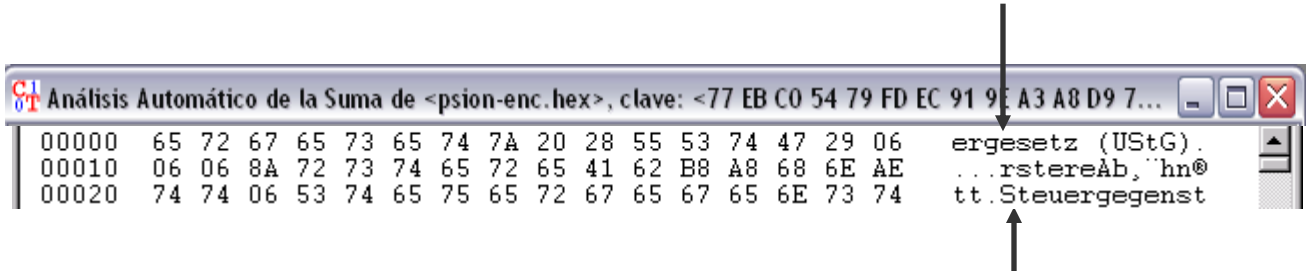
- Determinar las posiciones de bytes incorrectos, p.ej. 0xAA en la posición 3
- Adivinar y escribir los bytes correctos: “e” = 0x65

En un archivo inicial hex dump cifrado

- Determinar los bytes iniciales de las posiciones de bytes calculadas: 0x99
- Calcular los bytes correctos de la clave con CALC.EXE: 0x99 - 0x65 = 0x34

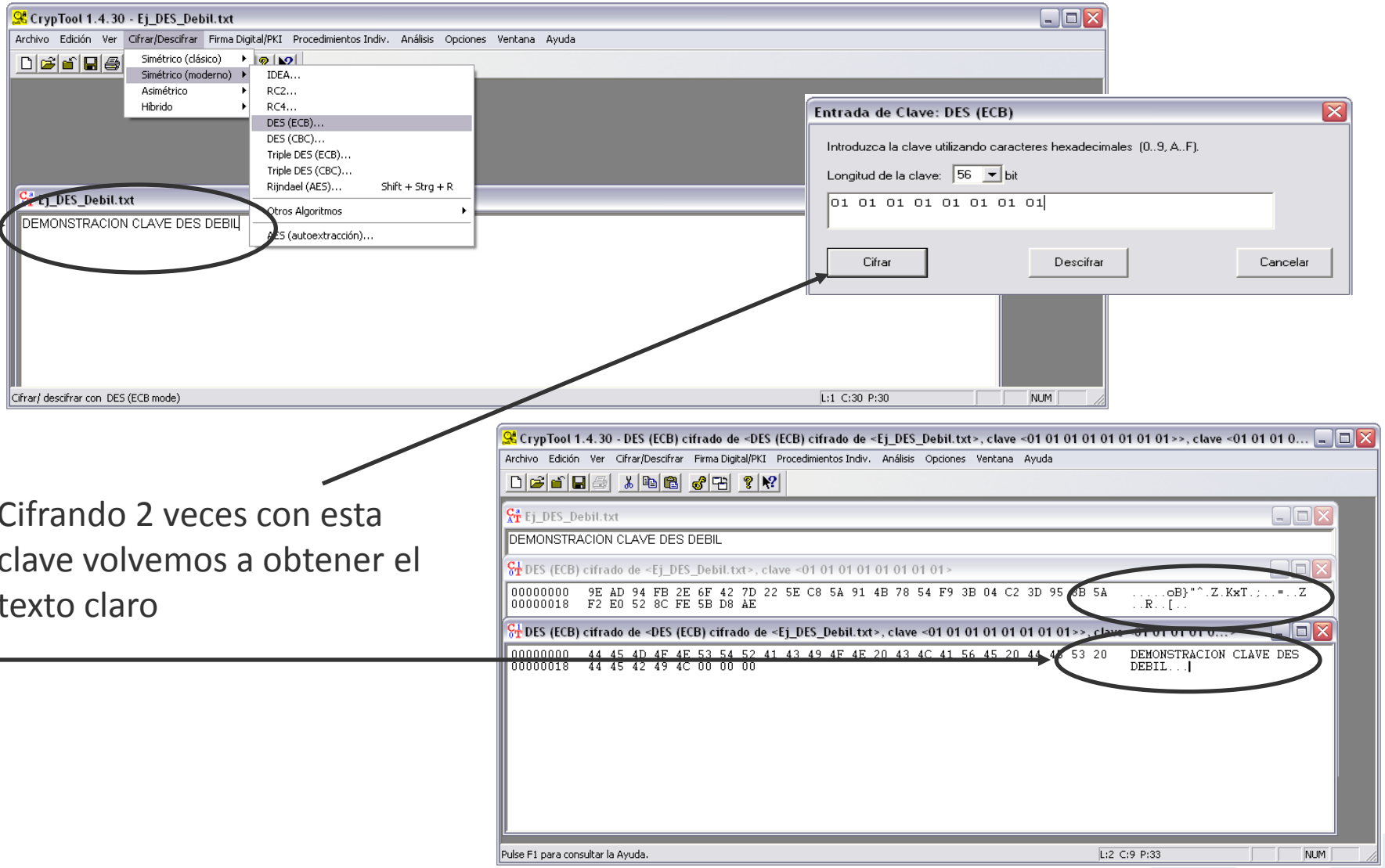
Clave desde el portapapeles

- Corrección 12865B**34**1498872C393E43741396A45670235E111E907AB7C0841...
- Descifrar el documento inicial cifrado utilizando la suma binaria
- bytes en la posición 3, 3+32, 3+2*32, ... Ahora son correctos



Ejemplos (5)

clave DES débil



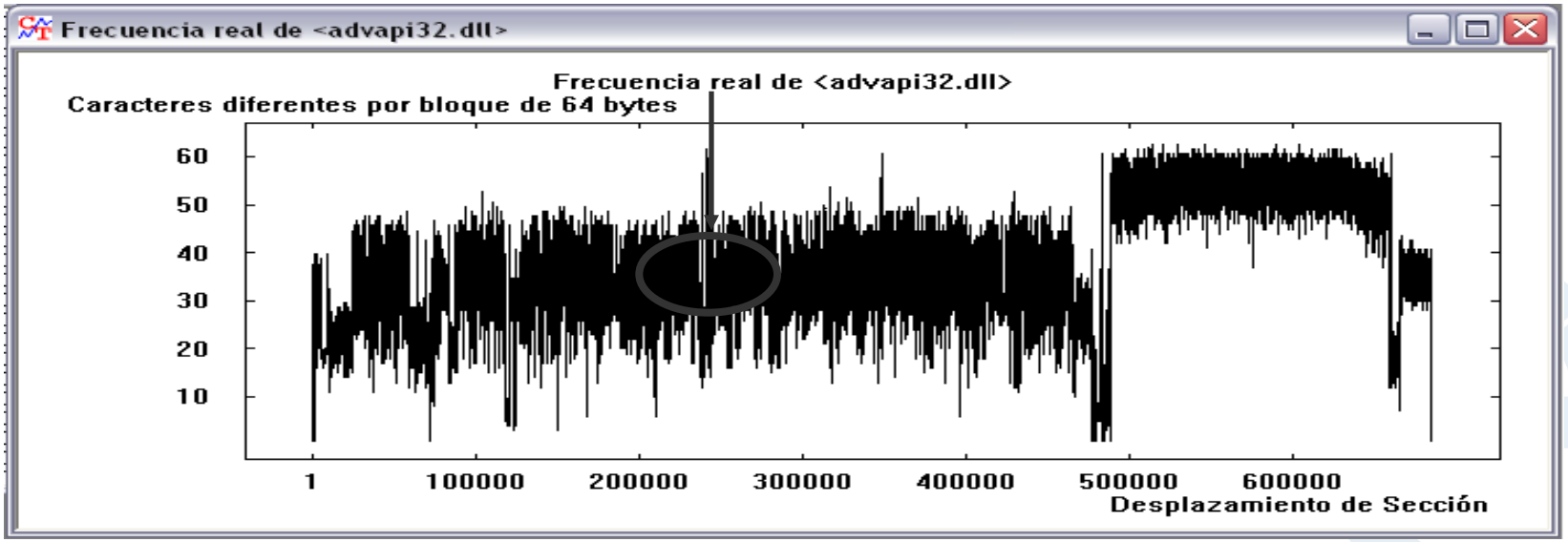
Ejemplos (6)

Localizar información de la clave

La función “Frecuencia real” es adecuada para localizar la información sobre la clave y áreas cifradas en archivos.

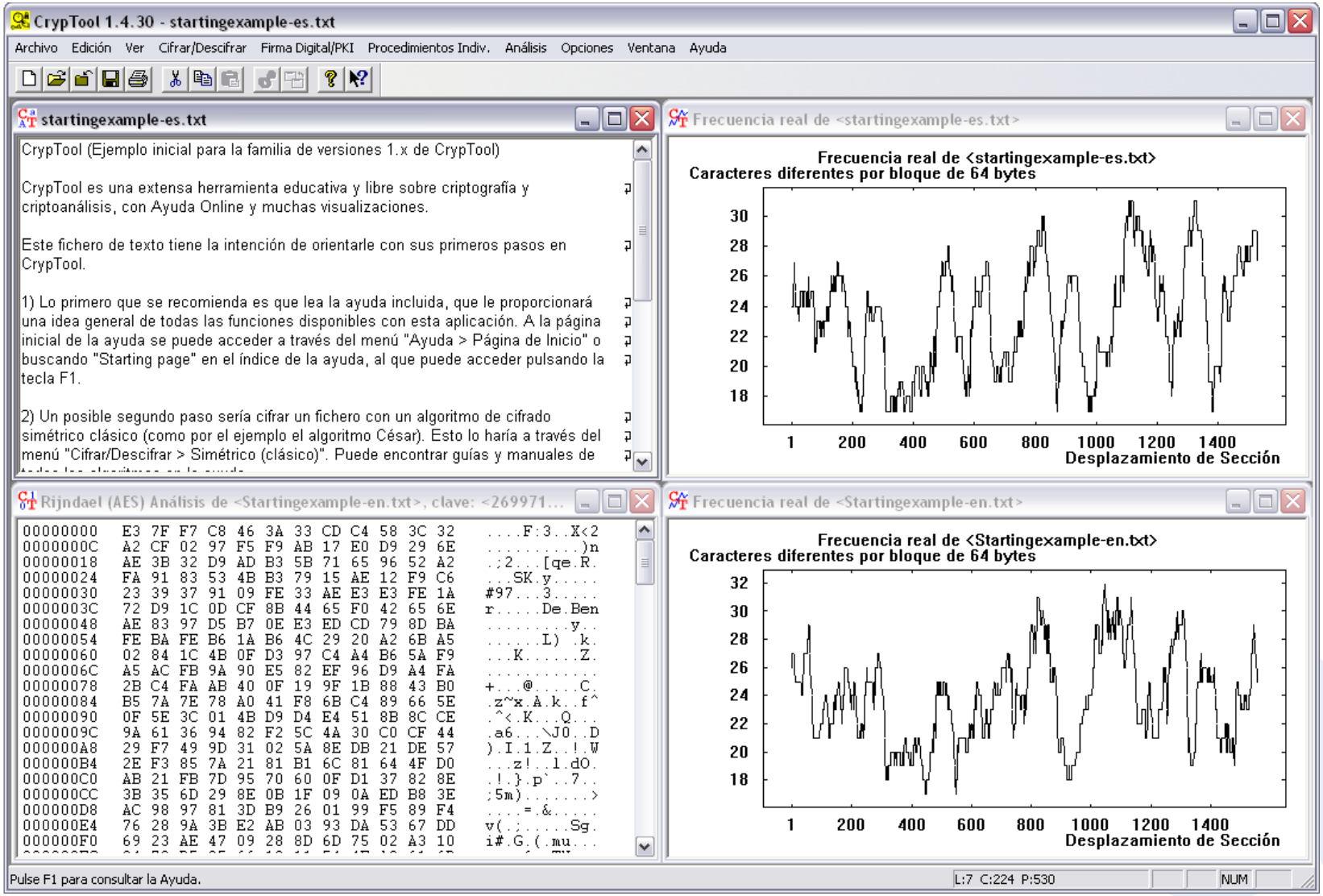
Trasfondo:

- Los datos de la clave son “más aleatorios” que el texto o el código programado
- Se puede reconocer como los picos en la ”frecuencia real”
- Ejemplo: la “clave NSA” de advapi32.dll (Windows NT)



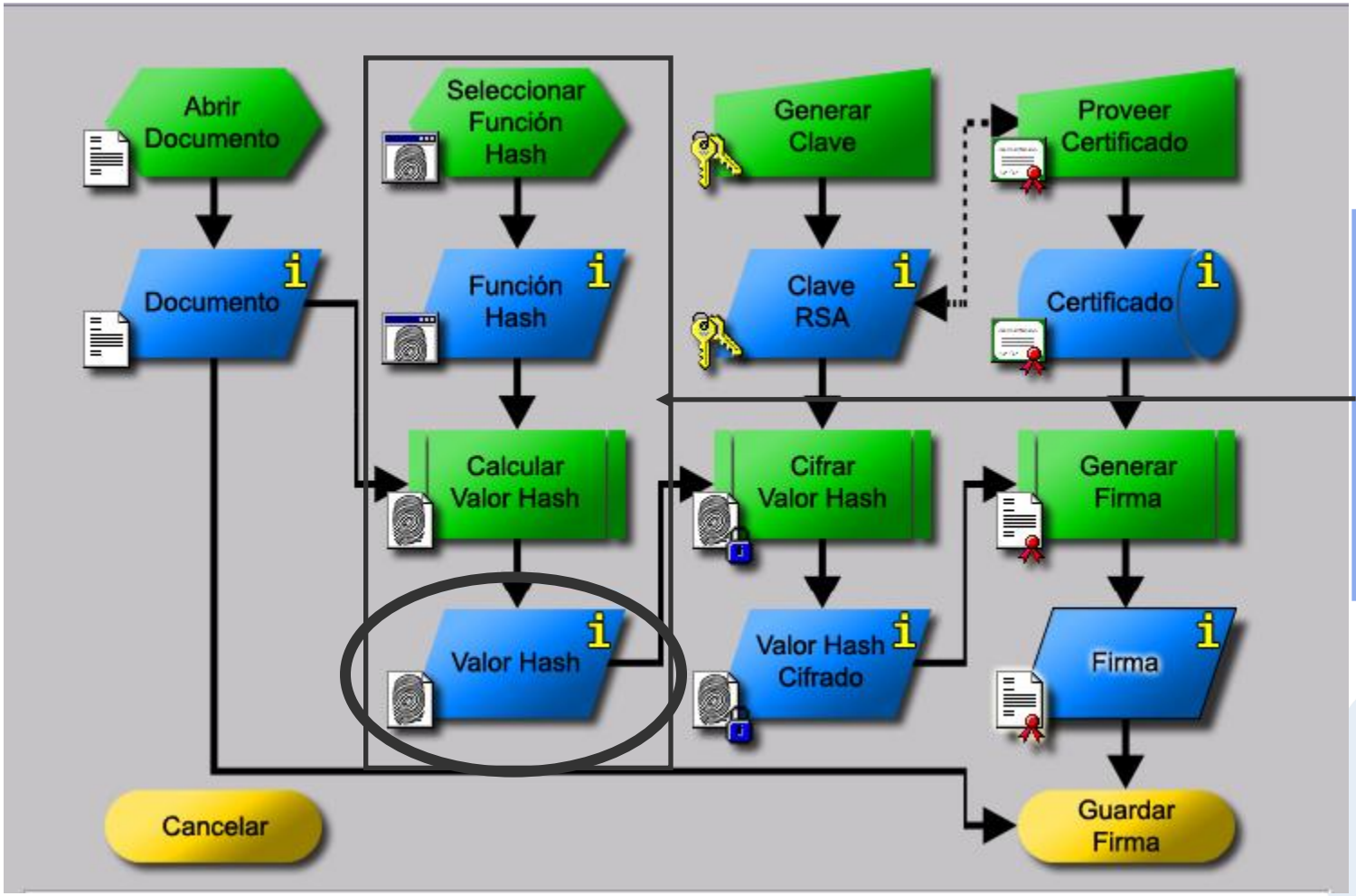
Ejemplos (6)

Comparación de frecuencia real con otros archivos



Ejemplos (7)

Ataque a la firma digital



Ataque:

¡Encontrar dos mensajes con el mismo valor hash!

Menú: “Análisis” \ “Hash” \ “Ataque al valor Hash de una firma digital”

Ejemplos (7)

Ataque a la firma digital– idea (I)

Ataque a la firma digital de un texto en ASCII basado en la búsqueda de colisiones.

Idea:

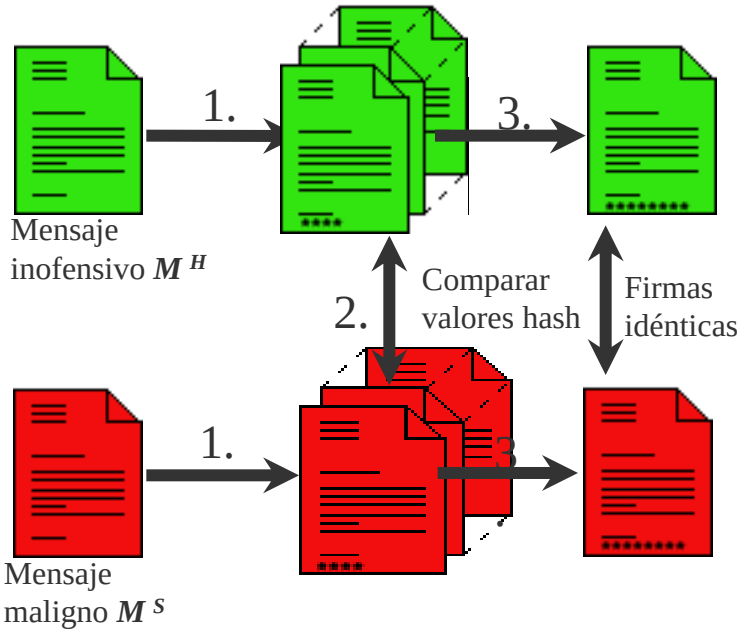
- Los textos ASCII se pueden modificar cambiando/insertando caracteres **no imprimibles**, sin cambiar el contenido visible
- Modificar dos textos en paralelo hasta encontrar una colisión hash
- Aprovechar la paradoja del cumpleaños (ataque del cumpleaños)
- Ataque genérico aplicable también a las funciones hash
- Se puede ejecutar en paralelo en varias máquinas (no está implementado)
- Se ha implementado en CrypTool como parte de la tesis de un licenciado "*Métodos y herramientas para ataques a firmas digitales*" (alemán), 2003.

Conceptos:

- Mapeados
- Algoritmo de Floyd modificado (consumo de memoria constante)

Ejemplos (7)

Ataque a la firma digital– idea (II)



1. **Modificación**: empezando desde un mensaje *M* se crean N mensajes distintos $M_1, ..., M_N$ con el mismo "contenido" como *M*.
2. **Búsqueda:** encontrar mensajes modificados M_i^H y M_j^S con el mismo valor hash.
3. **Ataque:** las firmas de ambos documentos M_i^H y M_j^S son iguales.

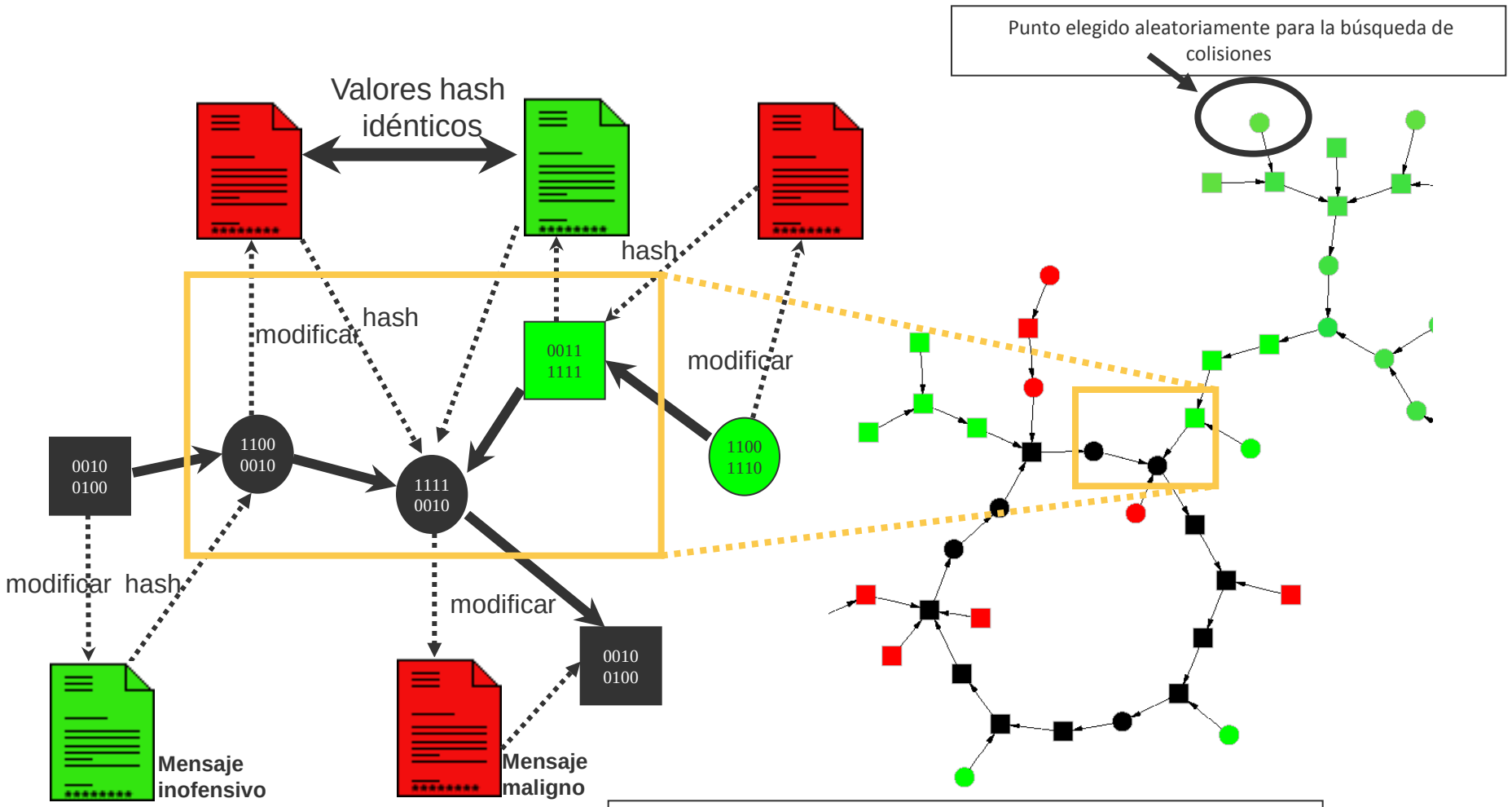
Conocemos por la paradoja del cumpleaños que para valores hash de longitud n en bits:

- buscar colisiones entre M^H y $M_1^S, ..., M_N^S$: $N \approx 2^n$
- Buscar colisiones entre $M_1^H, ..., M_N^H$ y $M_1^S, ..., M_N^S$: $N \approx 2^{n/2}$

Número estimado de mensajes generados para encontrar una colisión hash.

Localizar Colisiones Hash (1)

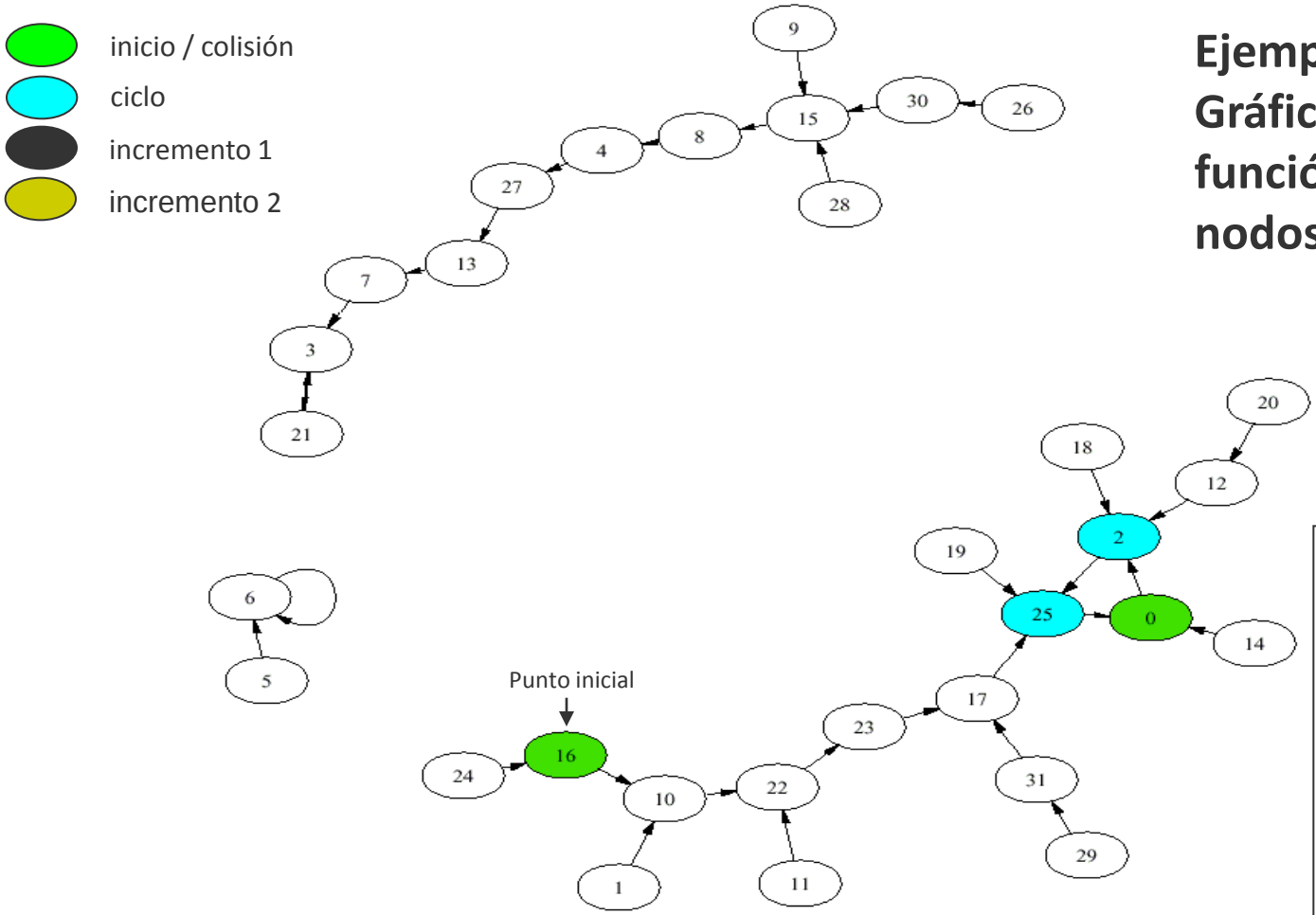
Mapeado por modificaciones del texto



- verde/ rojo: camino desde un árbol al ciclo – esto puede conducir a una colisión útil o inútil.
- cuadrado / círculo: valor hash par/impar
- negro: todos los nodos en el ciclo

Localizar Colisiones Hash (2)

Algoritmo de Floyd: encontrar el ciclo



Ejemplo: Gráfico de una función con 32 nodos

Paso 1: Localizar el punto que concuerde con el ciclo:

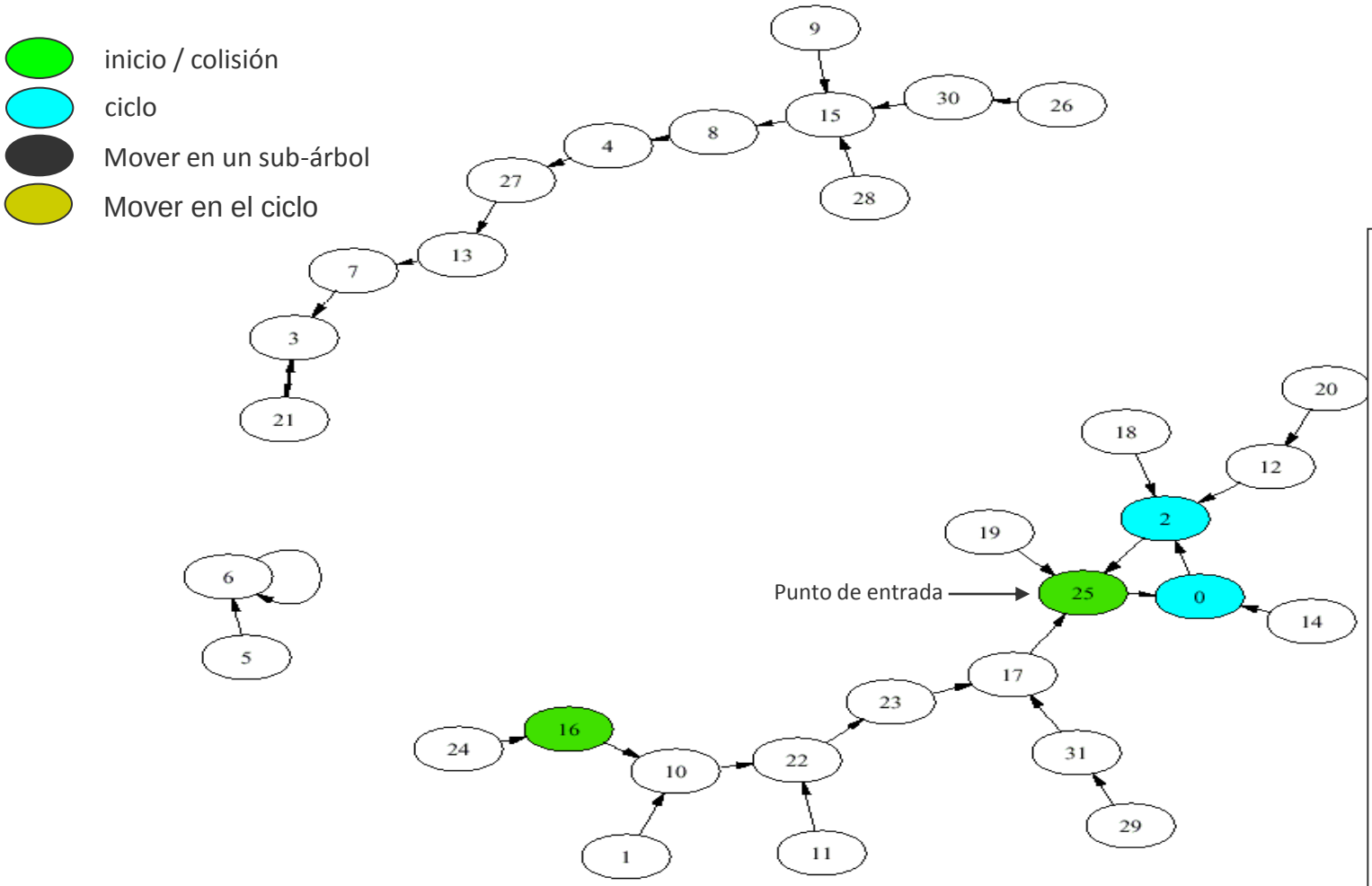
- Dos series con idéntico punto de inicio[16]:
  una serie con incremento 1, la otra con incremento 2.

Resultado (basado en teoría de grafos):

- Ambas series siempre terminan en un ciclo.
- Ambas series coinciden en un nodo en el ciclo (en este caso 0).

Localizar Colisiones Hash (3)

Seguir el ciclo (Extensión de Floyd): encontrar el punto de entrada



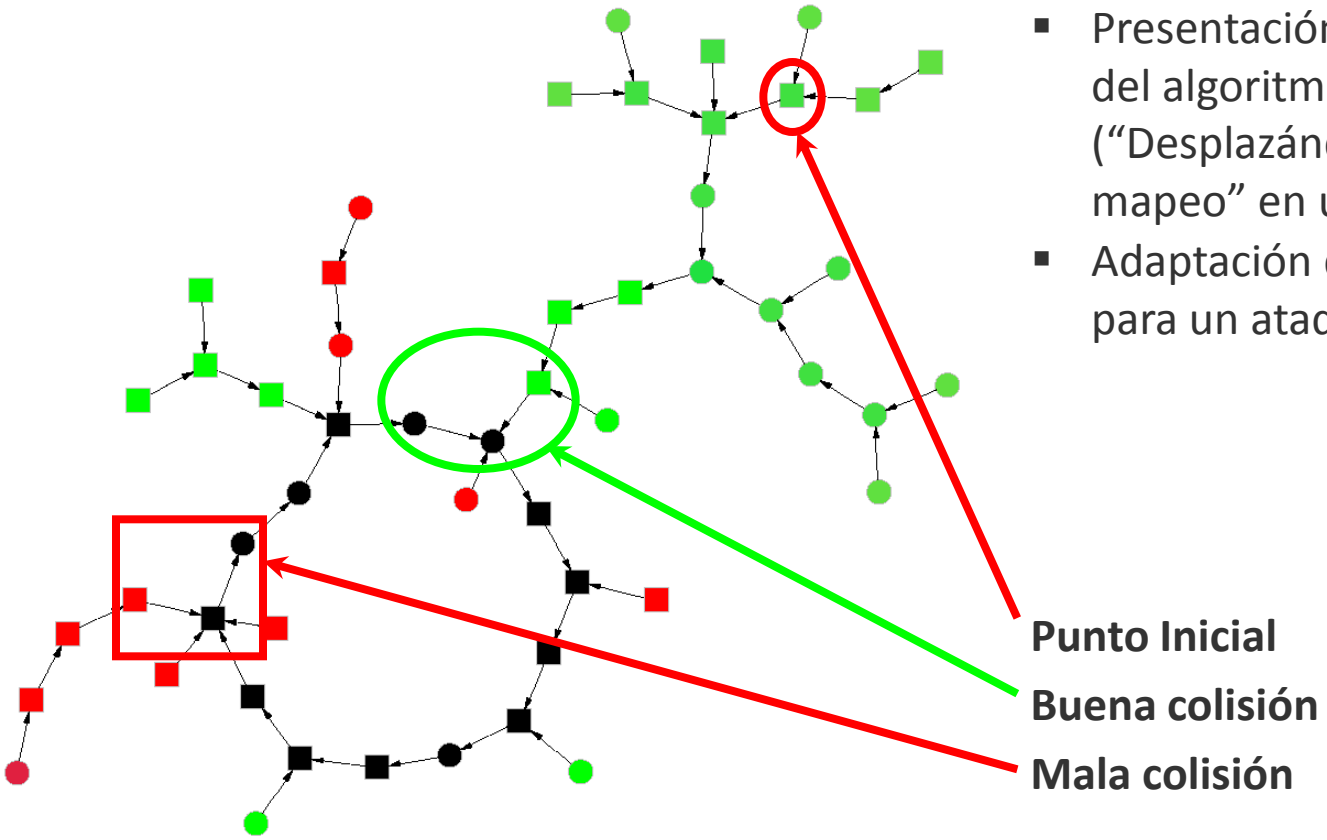
Paso 2: Localizar el punto de entrada de las series 1 en el ciclo [25]:

- La Serie 1 empieza otra vez desde el punto de entrada; la serie 3 con un incremento de 1 empieza en el munto de encuentro con el ciclo (en este caso en 0).

Resultado: Las series (1 y 3) coinciden en el punto de entrada del ciclo de la serie 1 (en este caso 25)

- Los predecesores (en estea caso 17 y 2) resultan en una colisión hash.

Ataque de la Paradoja del Cumpleaños a la Firma Digital



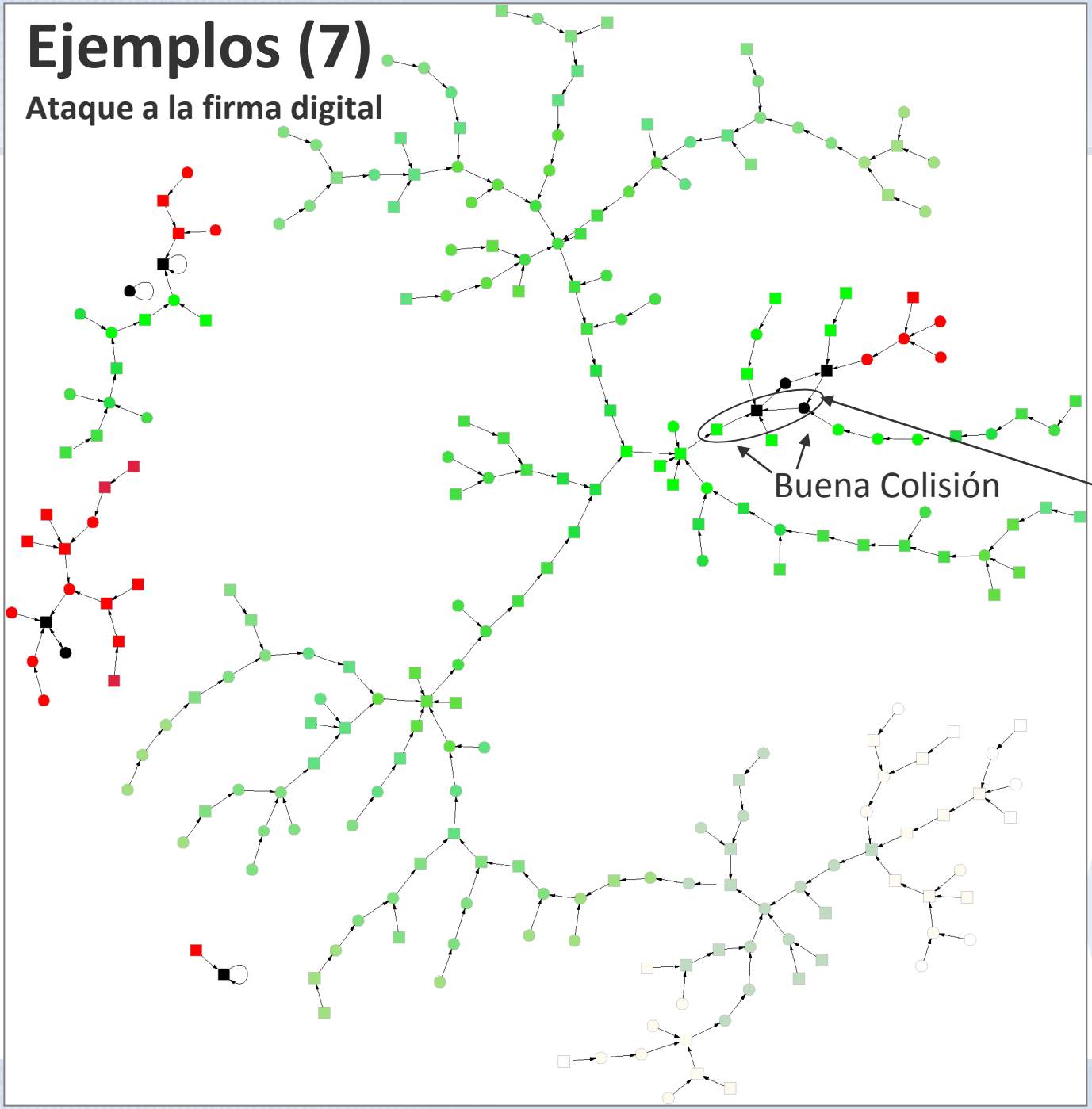
Examinar el algoritmo Floyd

- Presentación visual e interactiva del algoritmo Floyd ("Desplazándose a través del mapeo" en un ciclo).
- Adaptación del algoritmo de Floyd para un ataque de firma digital.

*El algoritmo de Floyd se implementa en CrypTool, pero la visualización del algoritmo aún no está implementada.

Ejemplos (7)

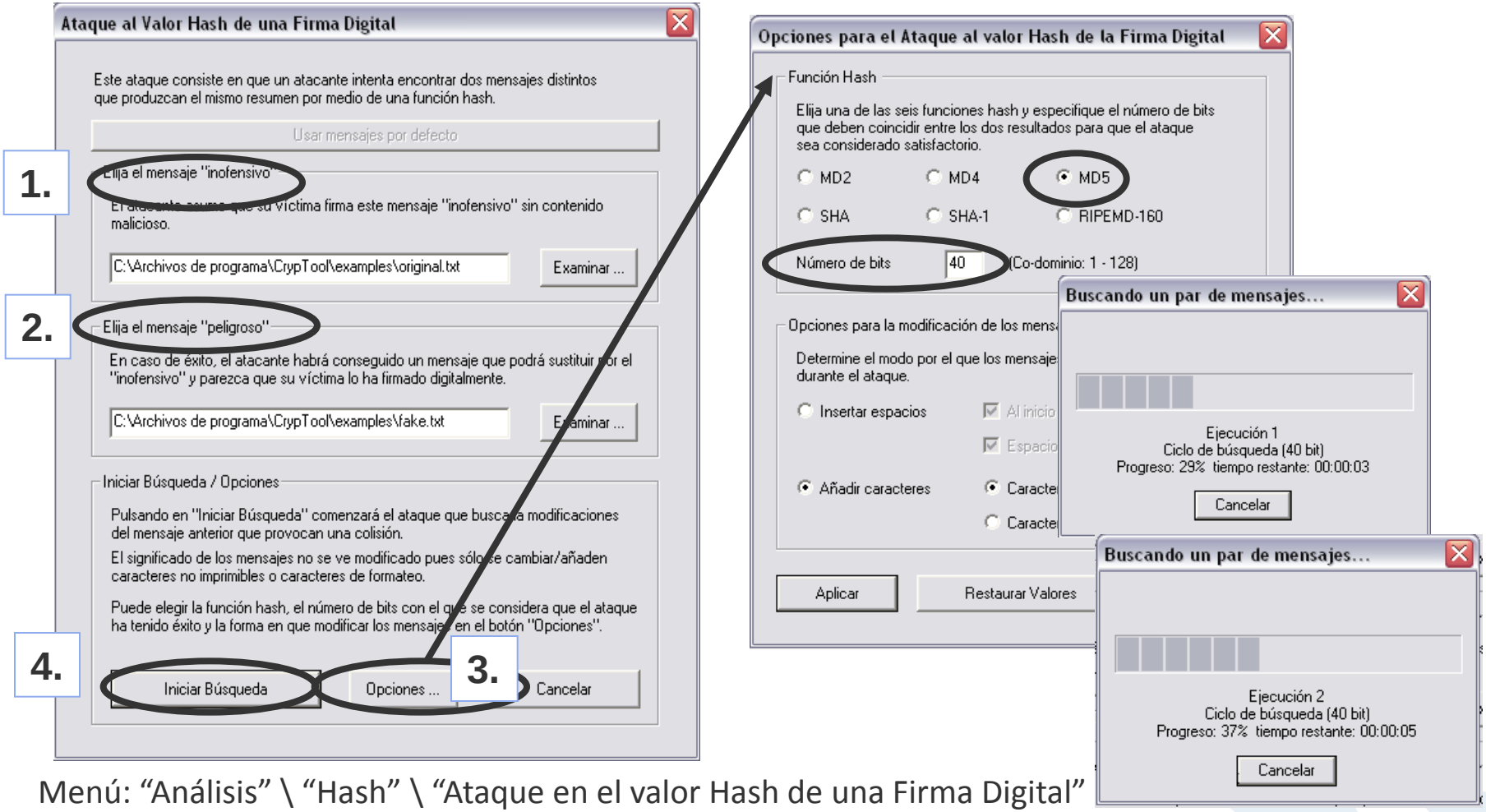
Ataque a la firma digital



Un ejemplo de un **"buen" Mapeado** (casi todos los nodos son verdes). En este grafo casi todos los nodos pertenecen al árbol grande, el cual se encuentra con el ciclo cuando se igualan los valores hash y donde el predecesor al punto de entrada en el ciclo es impar. Esto significa que el atacante encuentra útil la colisión para casi cualquier punto de inicio.

Ejemplos (7)

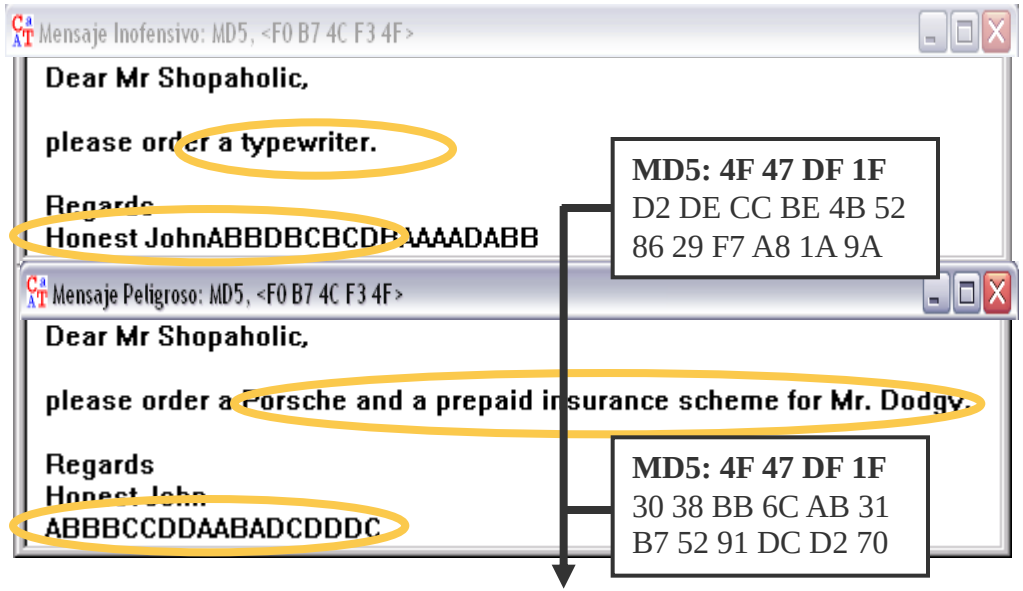
Ataque a la firma digital: Ataque



Menú: “Análisis” \ “Hash” \ “Ataque en el valor Hash de una Firma Digital”

Ejemplos (7)

Ataque a la firma digital: Resultados



Los primeros 32 bits de los valores hash son idénticos.

Resultados Experimentales

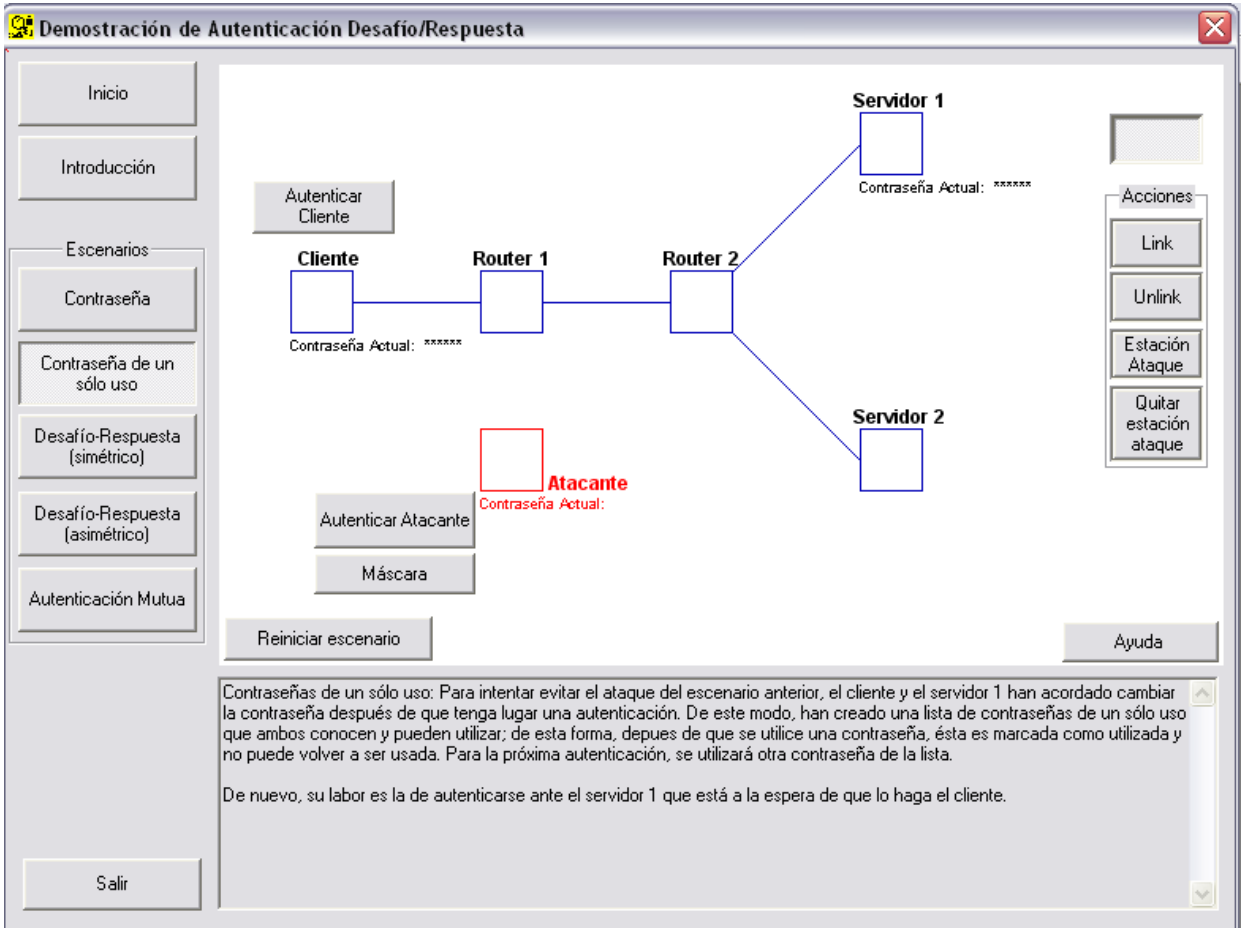
- *Colisión parcial de* 72 Bit (igualdad para los valores has de los primeros 72 bits) se encontró en un par de días en un único PC.
- ¡Las firmas que utilizan valores hash de hasta 128 bits se pueden atacar hoy en día con búsqueda en paralelo!
- Utilizar valores has de como poco 160bits de longitud.

Adicionalmente al manejo manual:

Característica automática desconectada en CrypTool: Ejecuta y registra los resultados para todos los conjuntos de configuraciones de parámetros. Disponible a través de la ejecución de CrypTool por línea de comandos.

Ejemplos (8)

Autentificación en un entorno cliente-servidor

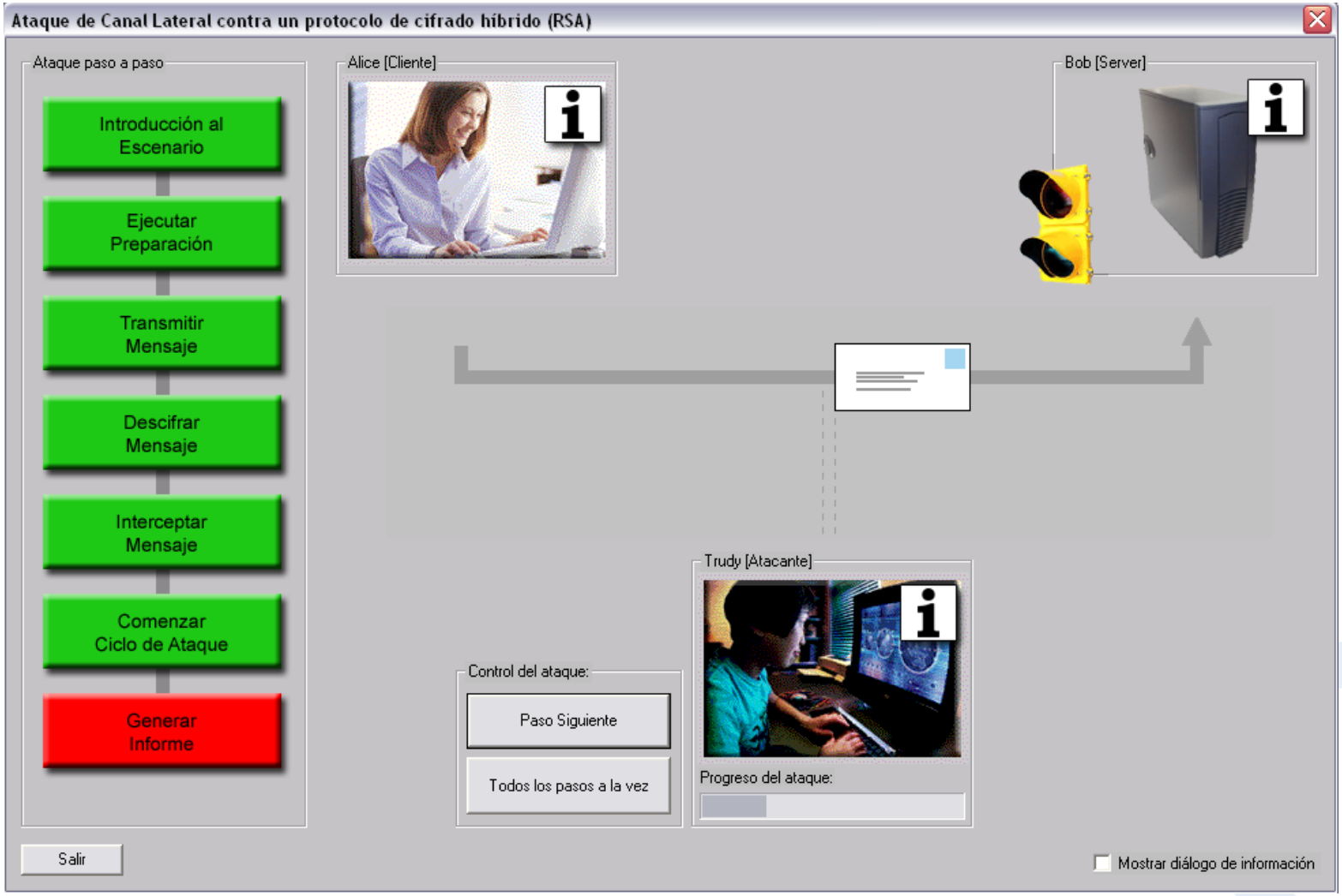


- Demostración interactiva para distintos métodos de autentificación.
- Oportunidades definidas del atacante.
- Puede tomar el papel del atacante.
- **Moraleja:**
  Sólo es segura la autenticación mutua.

Menú: “Procedimientos Indiv.” \ “Protocolos” \ “Autenticación en Red”

Ejemplos (9)

Demonstración de un ataque de canal lateral (en un protocolo de cifrado híbrido)



Menú: “Análisis” \ “Cifrado Asimétrico” \ “Ataque de canal lateral”

Ejemplos (9)

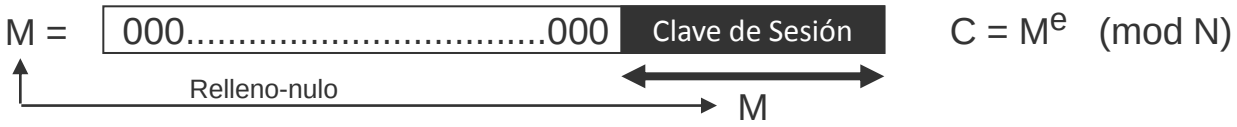
Idea para este ataque de canal lateral

Ulrich Kühn "*Side-channel attacks on textbook RSA and ElGamal encryption*"(2003)

Prerrequisitos:

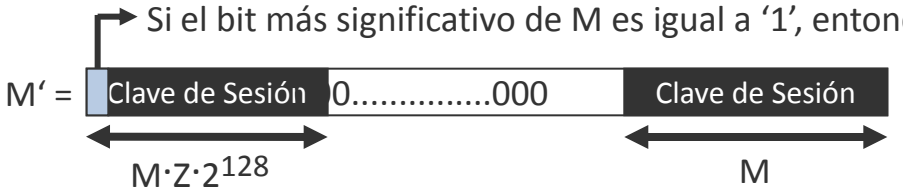
- Cifrado RSA: $C = M^e \pmod N$ y descifrado: $M = C^d \bmod N$.
- Las Claves de sesión de 128-Bits (en M) están "cifradas por diccionario" (relleno nulo).
- El servidor conoce la clave secreta d y
  - La utiliza después de descifrar sólo los 128 bits menos significativos (sin validación de los bits 0 de relleno) (esto significa que el servidor no reconoce si hay algo distinto a cero).
  - Avisos y mensajes de error, si del intento de cifrado resulta una clave de sesión errónea (el texto descifrado no se puede interpretar en el servidor). En el resto de casos no habrá mensajes.

Idea para el ataque: Aproximación para Z en la ecuación N = M * Z para cada M = ||N/Z||



$C = M^e \pmod N$

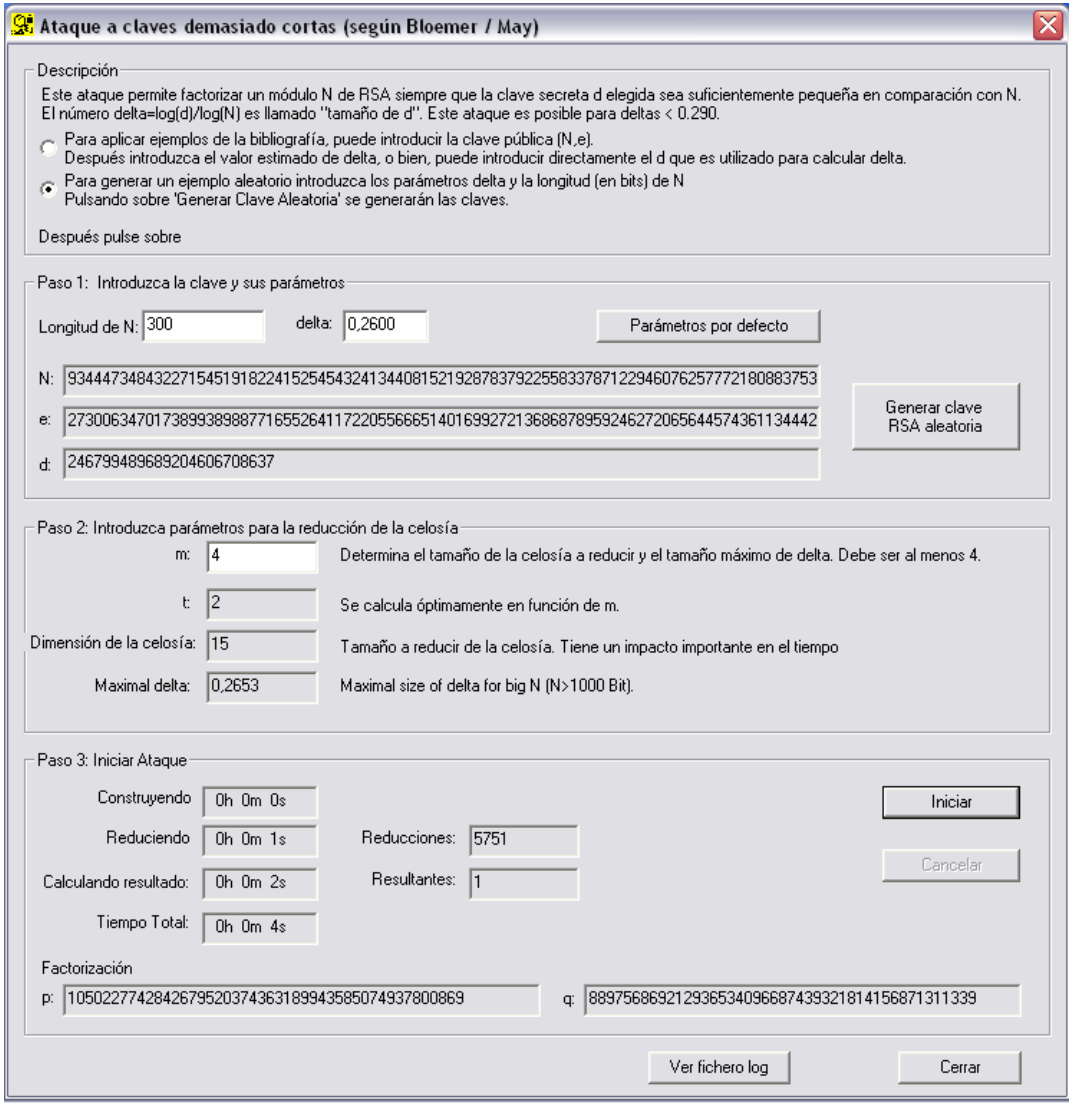
Se calculan de forma sucesiva todas las posiciones de bits para Z: En cada paso se toma un bit más. El atacante modifica C a C' (ver más abajo). Si ocurre un desbordamiento de bits mientras se calcula M' en el servidor (receptor), el servidor envía un mensaje de error. Basándose en esta información, el atacante obtiene un bit para Z.



$C' = M'^e = M^e \cdot (1+Z \cdot 2^{128})^e \pmod N$

Ejemplos (10)

Matemáticas: Ataques a RSA utilizando reducción de retículos (Lattice Reduction)



- Muestra cómo se tienen que elegir los parámetros del método RSA, por eso el algoritmo resiste al ataque de reducción de retículos descrito en la bibliografía.
- **3 variantes**
  1. El exponente secreto d es demasiado pequeño en comparación con N.
  2. Se conoce parcialmente a uno de los factores de N.
  3. Se conoce una parte del texto claro.
- Estas suposiciones son realistas.

Menú: “Análisis” \
“Cifrado Asimétrico” \
“Colección de ataques basados en RSA” \ …

Ejemplos (11)

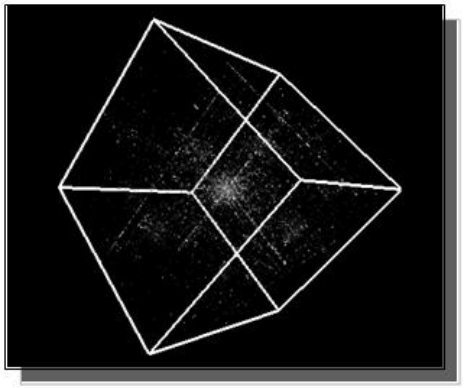
Análisis de Aleatoriedad con visualización 3-D

Puede girar el cubo con el mouse para obtener una mejor perspectiva.

Visualización 3D para el análisis de aleatoriedad

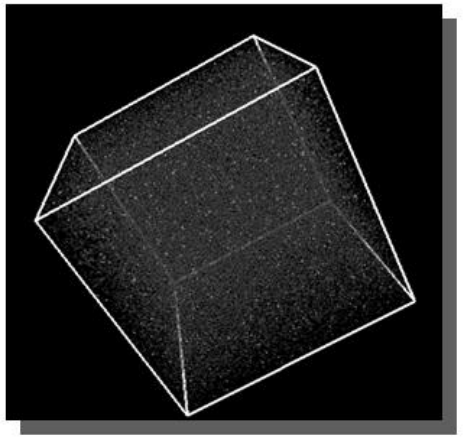
Ejemplo 1

- Abrir un archivo arbitrario (p.ej. Un informe en Word o una presentación de PowerPoint)
- Se recomienda seleccionar un archivo de al menos 100 Kb
- Análisis 3D
- Resultado: **se reconocen fácilmente las estructuras**



Ejemplo 2

- Generación de números aleatorios: “Procedimientos Indiv.” \ “Herramientas” \ “Generar Números Aleatorios”
- Se recomienda generar al menos 100.000 bytes aleatorios
- Análisis 3D
- Resultado: **distribución uniforme (no se reconocen las estructuras)**



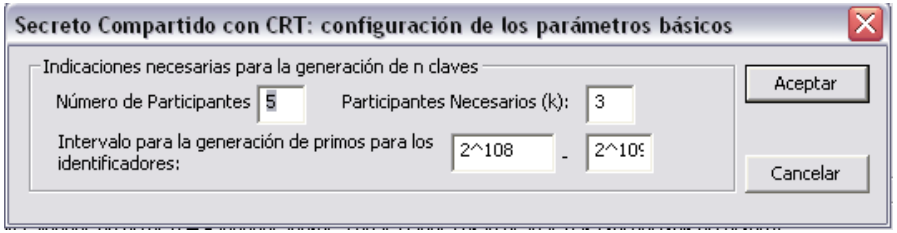
Menú: “Análisis” \ “Análisis de Aleatoriedad” \ “Visualización 3D”

Ejemplos (12)

Secreto Compartido con CRT – Implementación del Teorema Chino de los Restos (CRT)

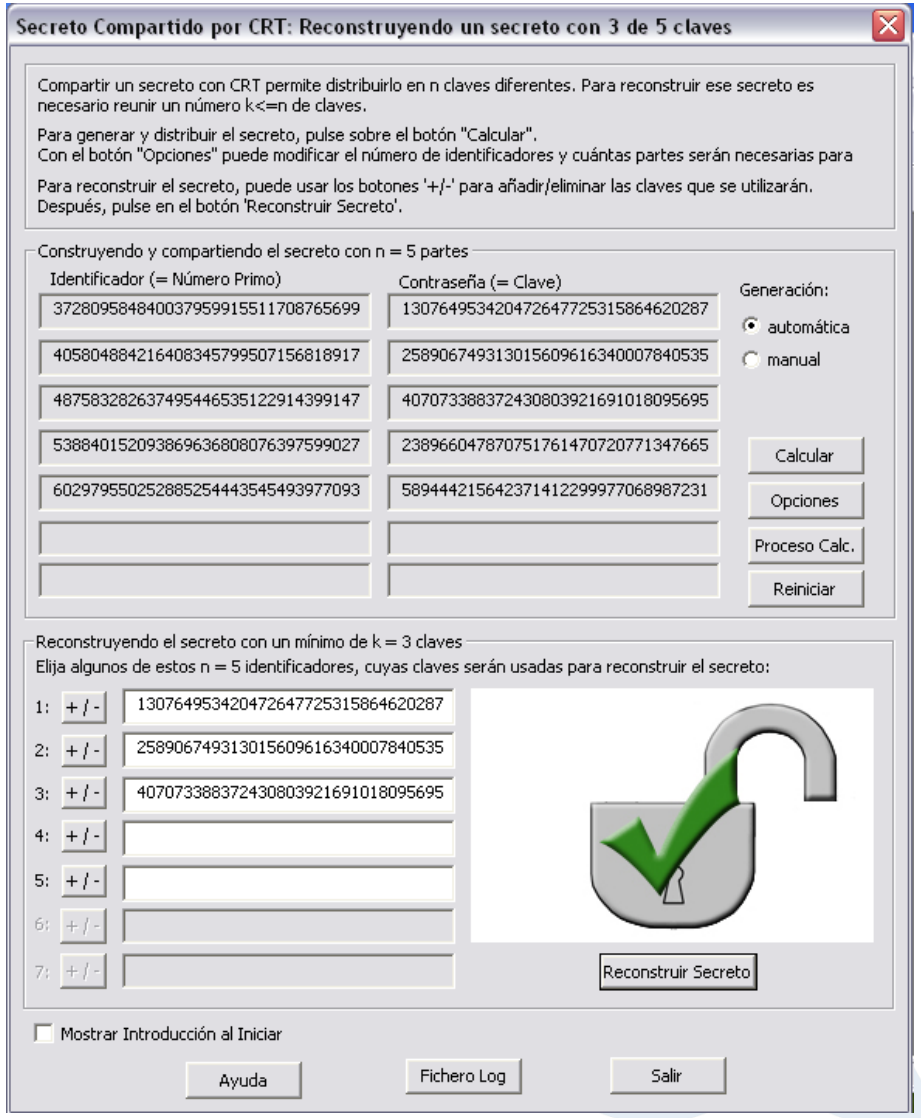
Ejemplo de Secreto compartido (1):

- **Problema:**
  - 5 personas tienen una clave
  - Para ganar acceso al menos 3 de esas 5 personas tienen que estar presentes
- **"Opciones"** permite configurar los detalles del método.



- **"Pasos de Calc."** muestra todos los pasos para generar la clave.

Menú: "Procedimientos Indiv." \
"Aplicaciones del Teorema Chino de los Restos"\
"Secreto Compartido por CRT"



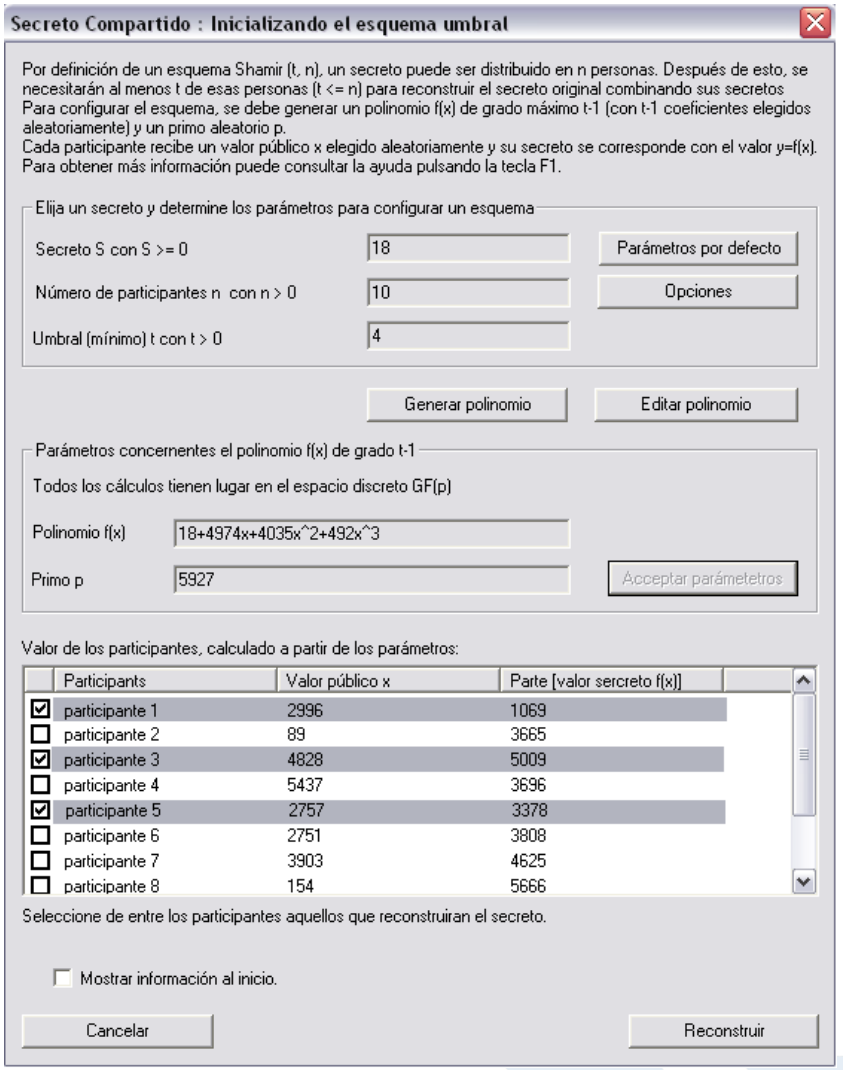
Ejemplos (12)

Secreto Compartido de Shamir

Ejemplo Secreto Compartido (2)

- **Problema**
  - Un valor secreto se puede separar para n personas.
  - t de las n personas se necesitan para recuperar el valor secreto K.
  - (t, n) esquema de borde
- **Realizar lo siguiente:**
  1. Introducir el secreto K, número de personas n y él umbral t
  2. Generar polinomio
  3. Utilizar Parámetros
  4. Utilizando **“Reconstrucción”** se puede recuperar el secreto

Menú: “Procedimientos Indiv.” \ “Demostración Secreto Compartido (Shamir)”

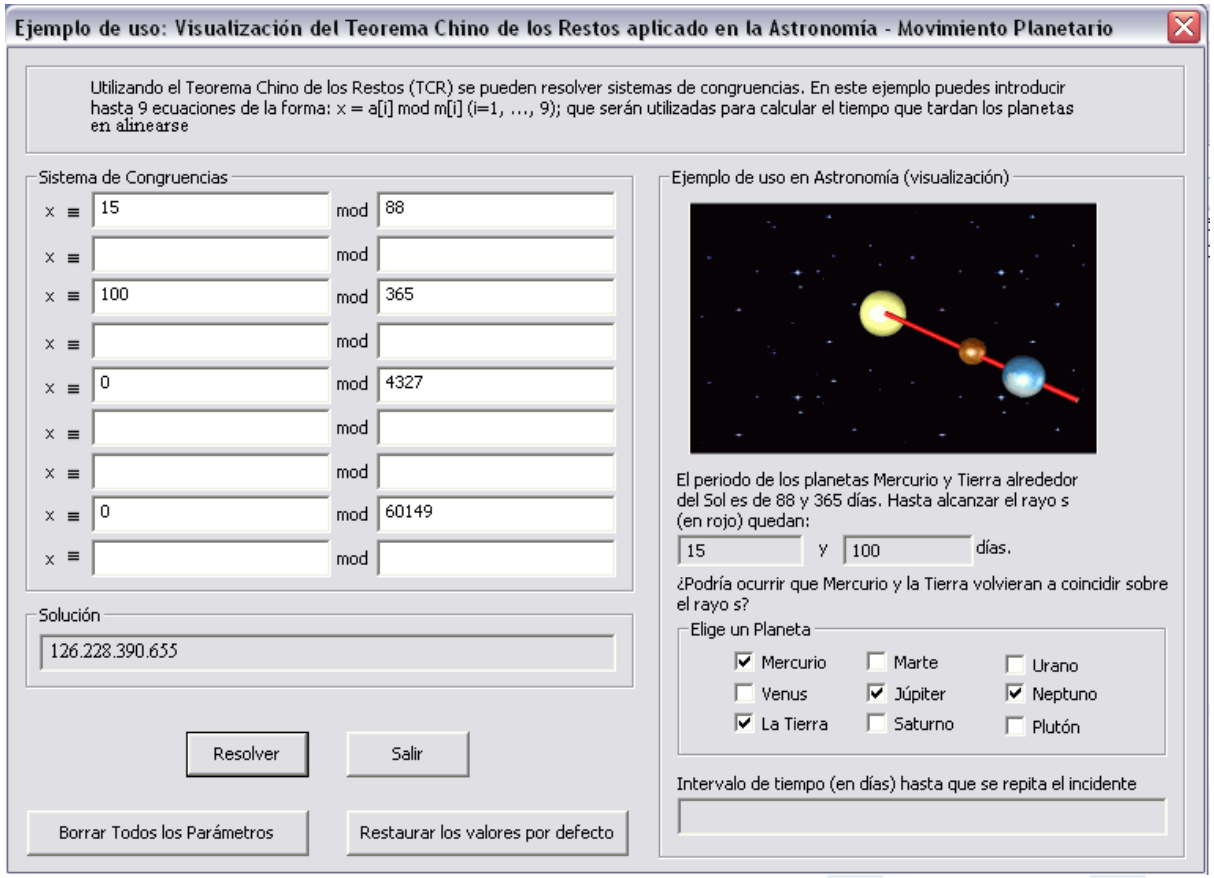


Ejemplos (13)

Implementación del CRT para resolver sistemas de ecuaciones modulares lineales

Escenario en astronomía

- ¿Cuánto tiempo tiene que pasar hasta que un número dado de planetas (con distintos períodos de rotación) se alineen?
- El resultado es un sistema lineal de ecuaciones modulares, que se puede resolver con el Teorema Chino de los Restos (CRT).
- En esta demostración se pueden introducir hasta 9 ecuaciones y calcular una solución utilizando el CRT.



Menú: "Procedimientos Indiv." \ "Aplicación del Teorema chino de los restos" \ "Astronomía y Movimiento planetario"

Ejemplos (14)

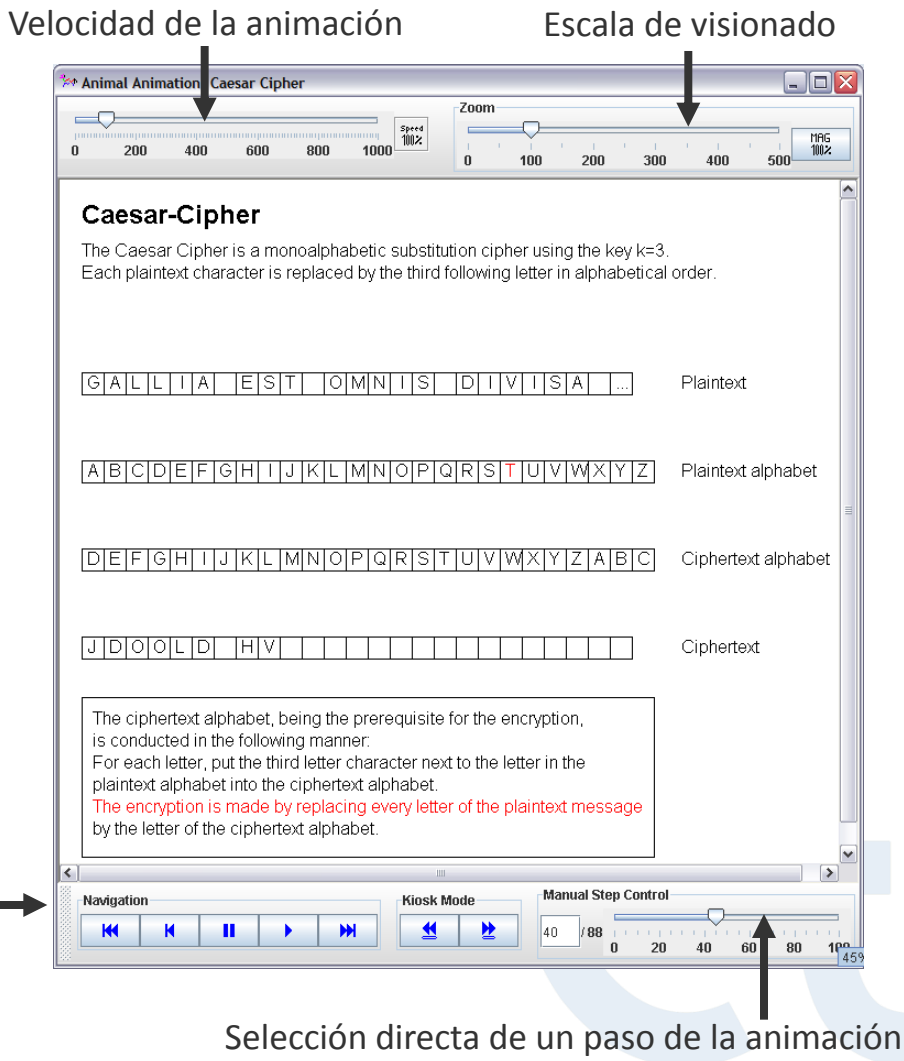
Visualización de métodos de cifrado simétrico utilizando ANIMAL (1)

Visualizaciones animadas de varios algoritmos simétricos

- César
- Vigenère
- Nihilist
- DES

CrypTool

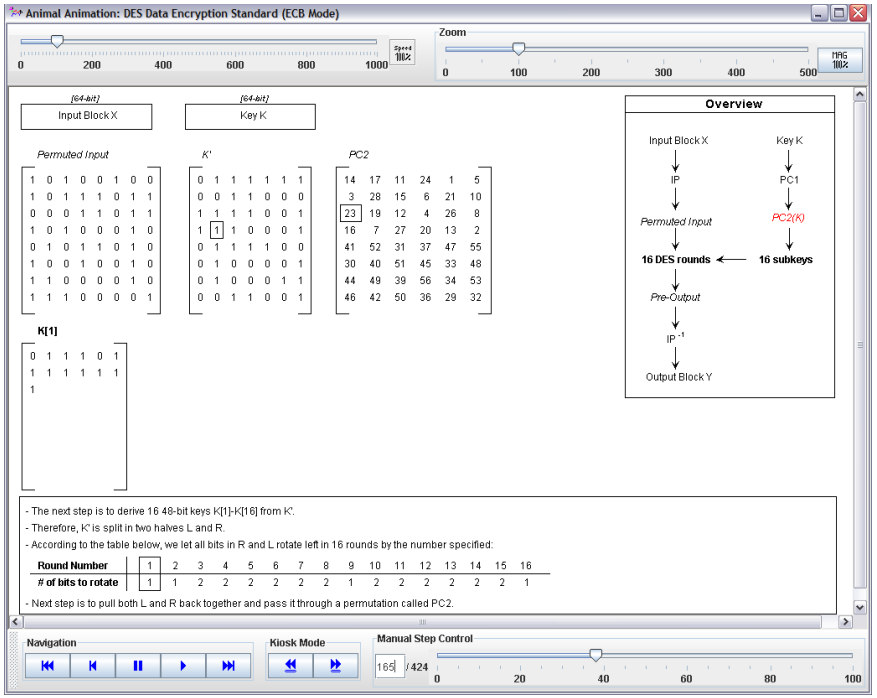
- Menú: “Procedimientos Indiv.” \ “Visualización de algoritmos” \ …
- Control de la animación interactivo utilizando los controles integrados en la ventana.

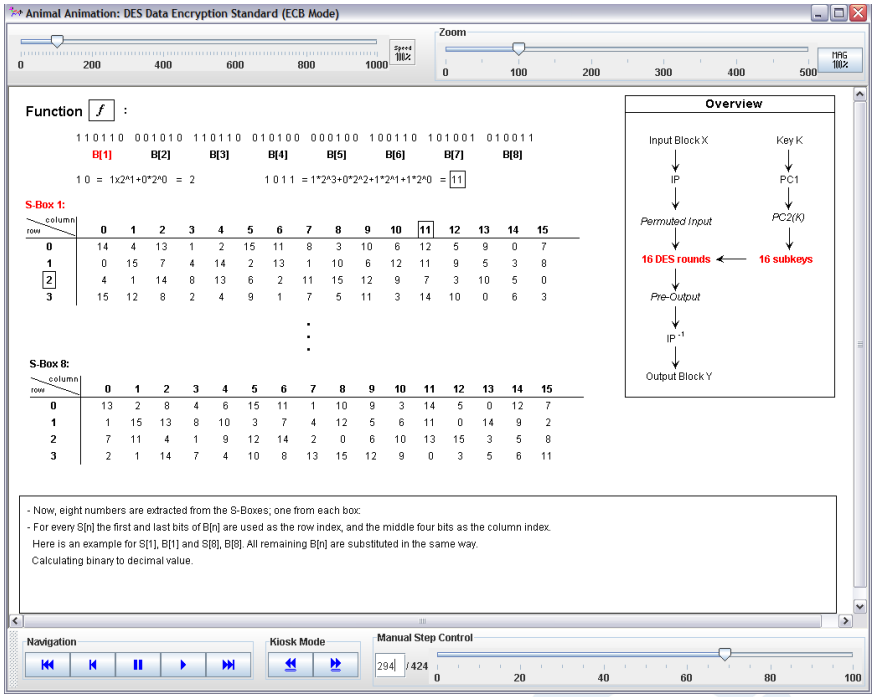


Ejemplos (14)

Visualización de métodos de cifrado simétrico utilizando ANIMAL (2)

Visualización del cifrado DES





Después de la permutación del bloque de entrada utilizando el vector de inicialización IV, la clave K se permuta con PC1 y PC2.

La función de núcleo *f* del DES, que se enlaza la mitad derecha del bloque R_{i-1} con la clave parcial K_i.

Ejemplos (15)

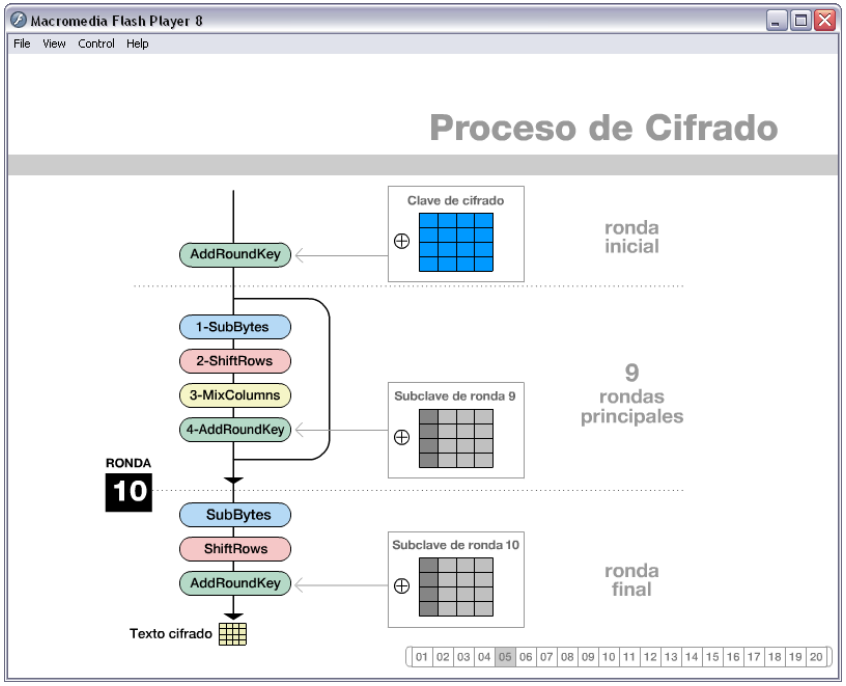
Visualización de AES (cifrado Rijndael)

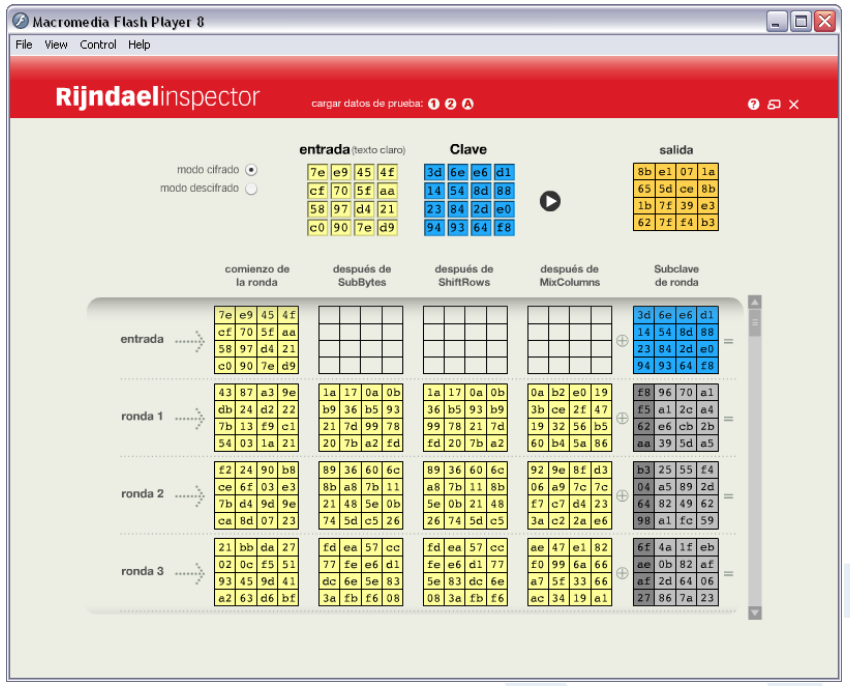
Animación Rijndael (el cifrado Rijndael fue el ganador de la dependencia del AES)

- La visualización muestra la animación del proceso del cifrado basado en ciclos (utilizando datos fijos)

Inspector Rijndael

- El proceso de cifrado para probar (utilizando tus propios datos)





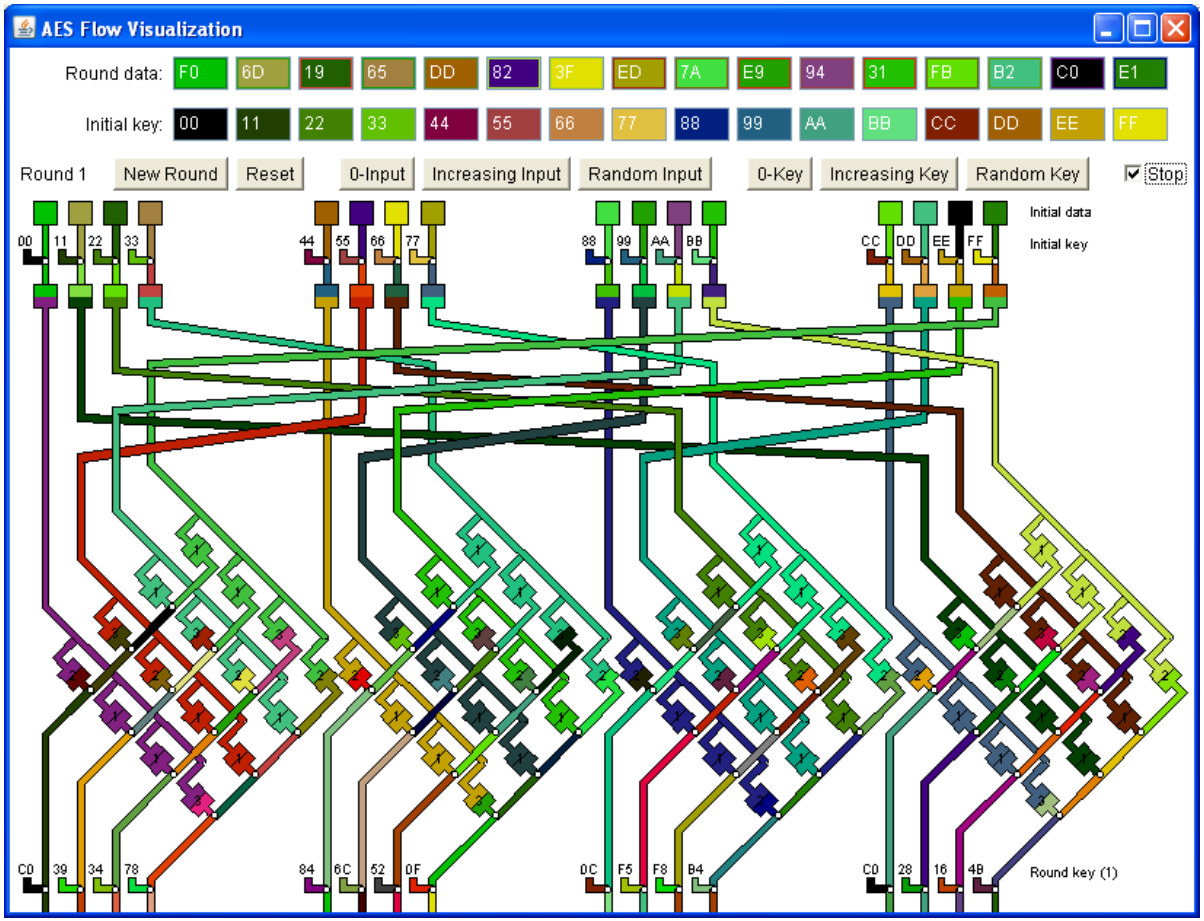
Menú: “Procedimientos Indiv.” \ “Visualización de Algoritmos” \ “AES” \ “Animación Rijndael” o “Inspector Rijndael”

Ejemplos (15)

Visualización de AES (cifrado Rijndael) – usando Java

Visualización de flujo de Rijndael

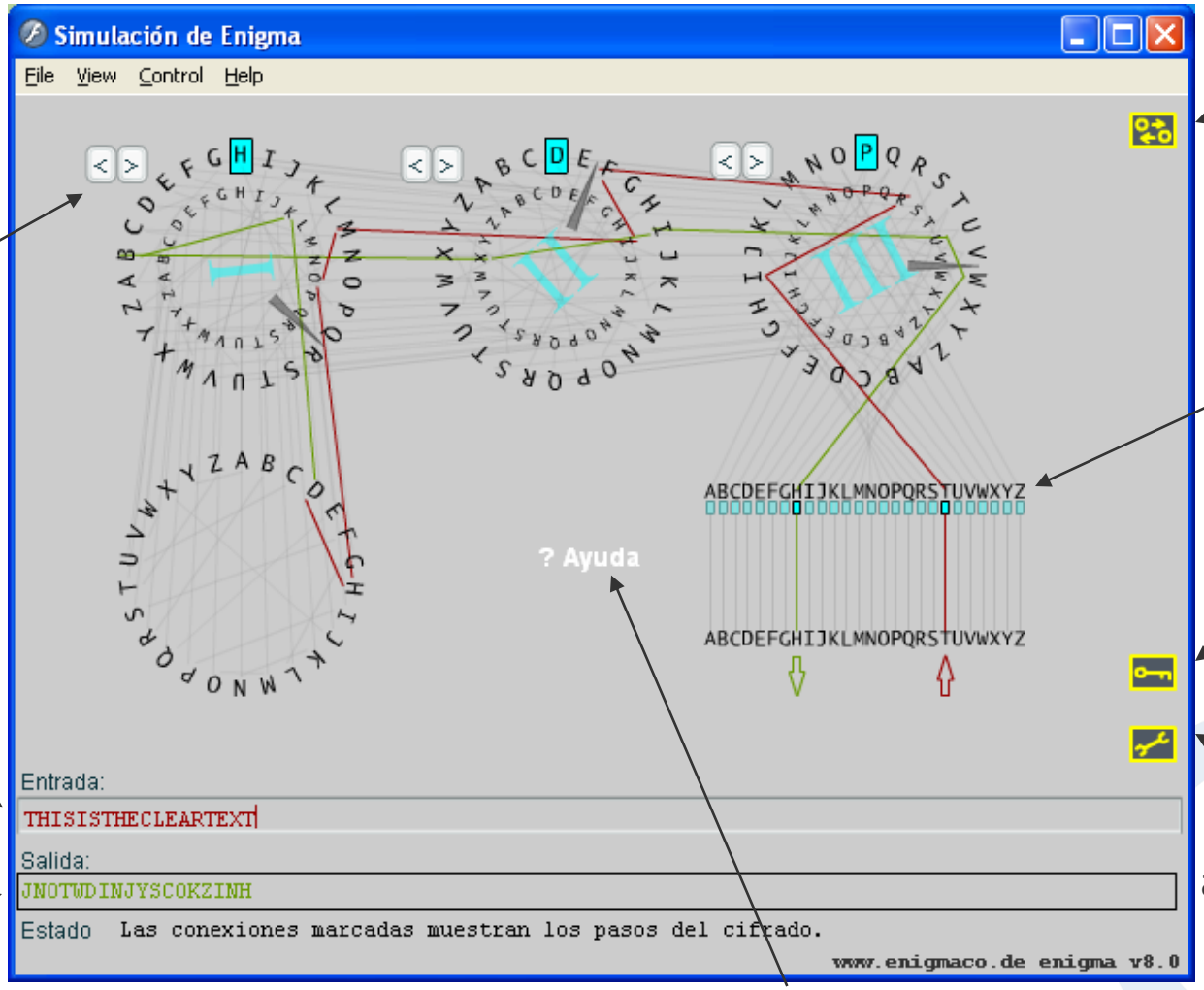
- Visualización de cambios por ronda a través de escala de colores.



Menú: “Procedimientos Indiv.” \ “Visualización de Algoritmos” \ “AES” \ “Animación Rijndael …”

Ejemplos (16)

Visualización del cifrado de Enigma



Seleccionar rotores

Cambiar configuración del rotor

Cambiar conectores

Mostrar configuración

Entrada del texto claro

Salida del texto cifrado

Reiniciar Enigma al estado inicial o a un estado aleatorio

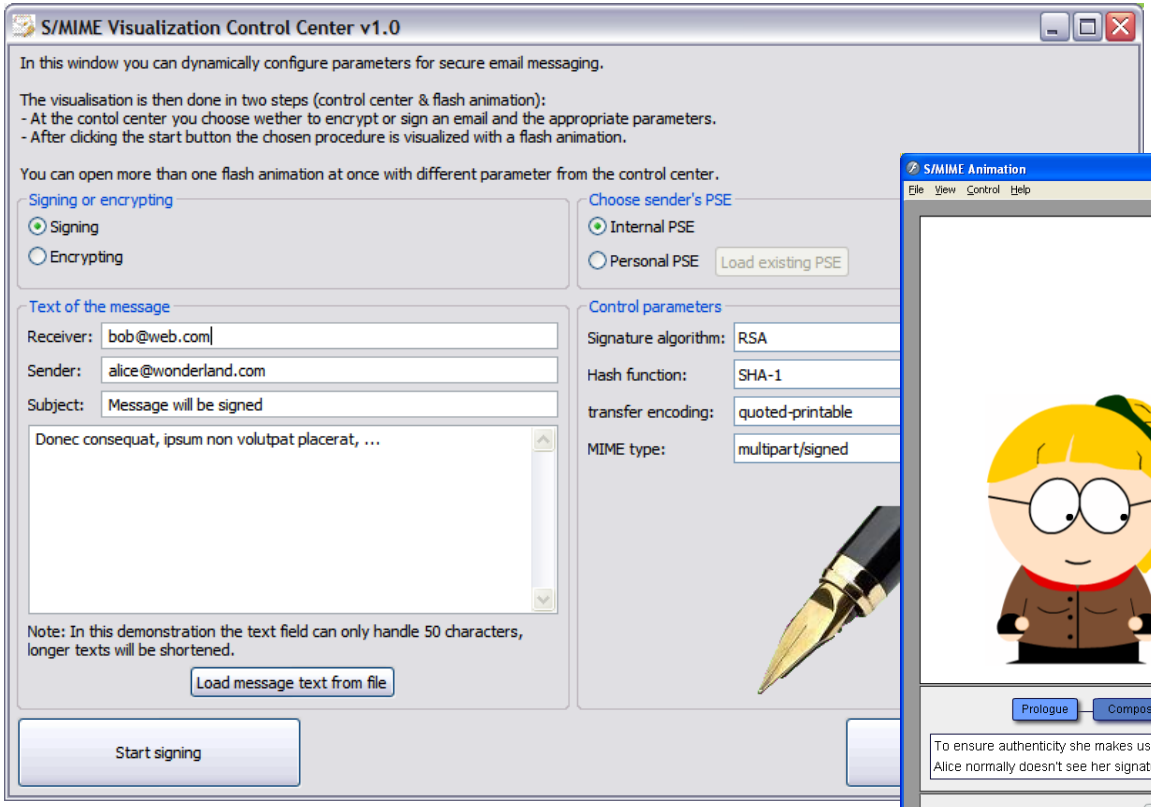
Ayuda en línea adicional HTML

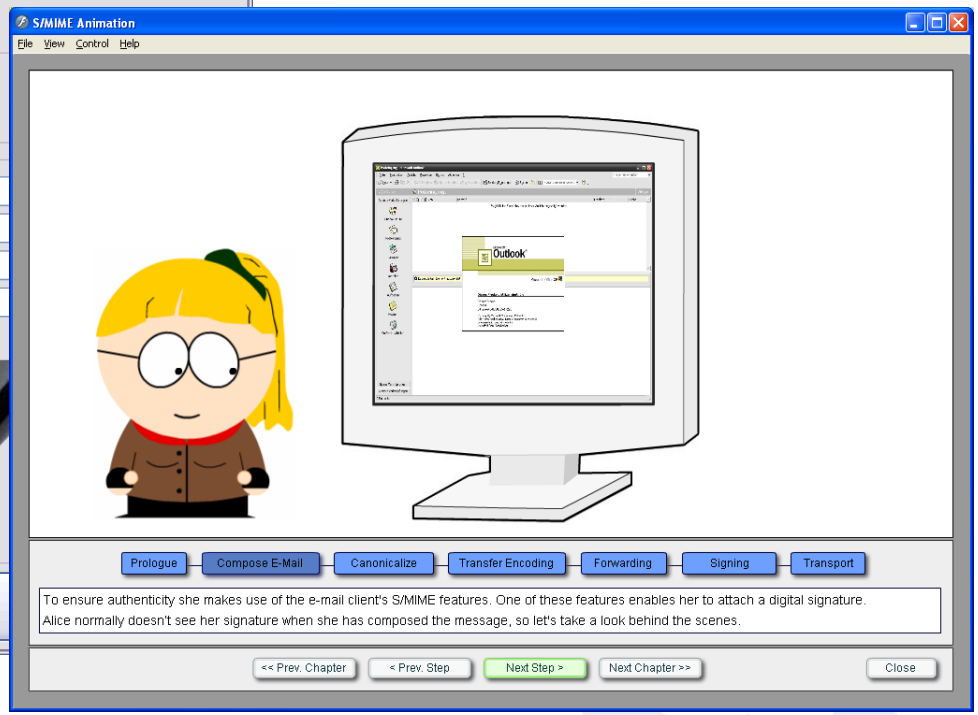
Ejemplos (17)

Visualización de E-Mail seguro usando S/MIME

Visualización S/MIME

- Centro de Control: Firmar/Cifrar mensajes con diferentes parámetros
- Animación: Desde la creación en el emisor hasta la lectura en el receptor





Menú: “Procedimientos Indiv.” \ “Protocolos“ \ “Seguridad E-Mail con S/MIME…“

Ejemplos (18)

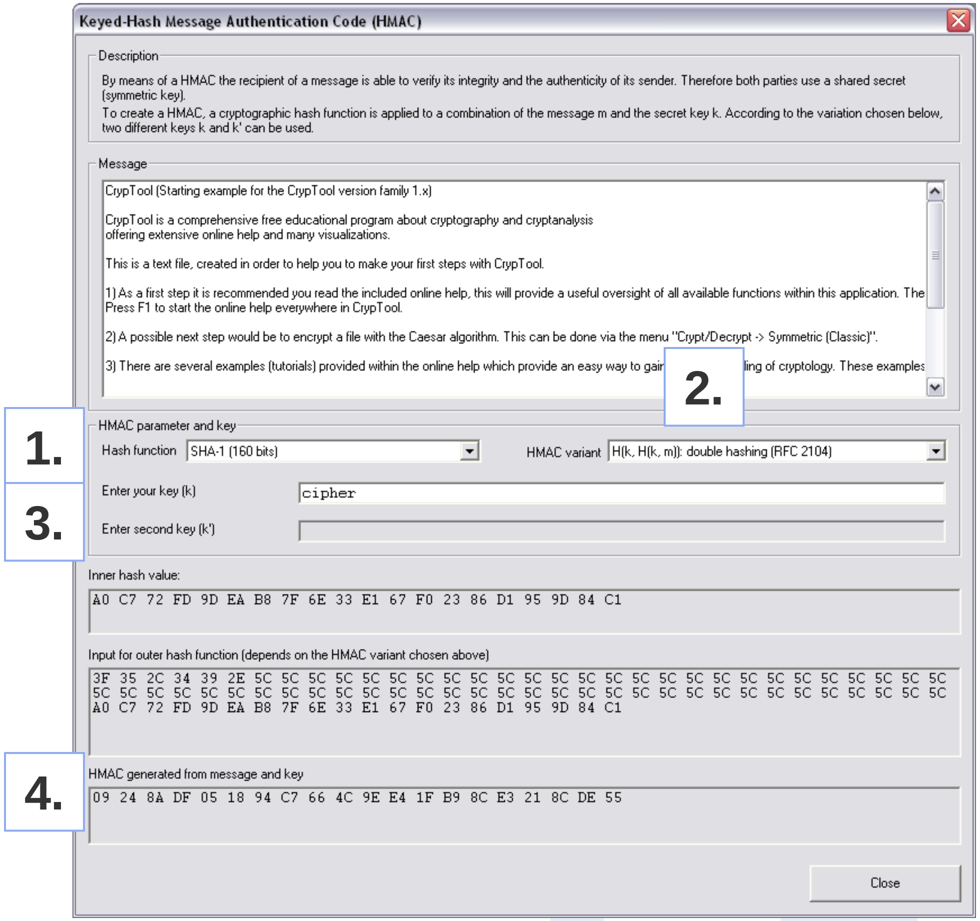
Generación de un código de autentificación de un mensaje (HMAC)

Código de Autentificación de Mensaje (HMAC)

- Asegura:
  - La integridad de un mensaje
  - La autenticación del mensaje
- Bases: una clave común
- Alternativa: Firma Digital

Generación de un MAC en CrypTool

1. Elija una función hash
2. Seleccione una variante de MAC
3. Introduzca una clave (dependiendo de la variante del MAC pueden ser dos claves)
4. Generación del MAC (automático)



Menú: “Procedimientos Indiv.” \ “Hash” \ “Generación de MACs”

Ejemplos (19)

Demostración Hash

Sensibilidad de las funciones hash a las modificaciones del texto claro

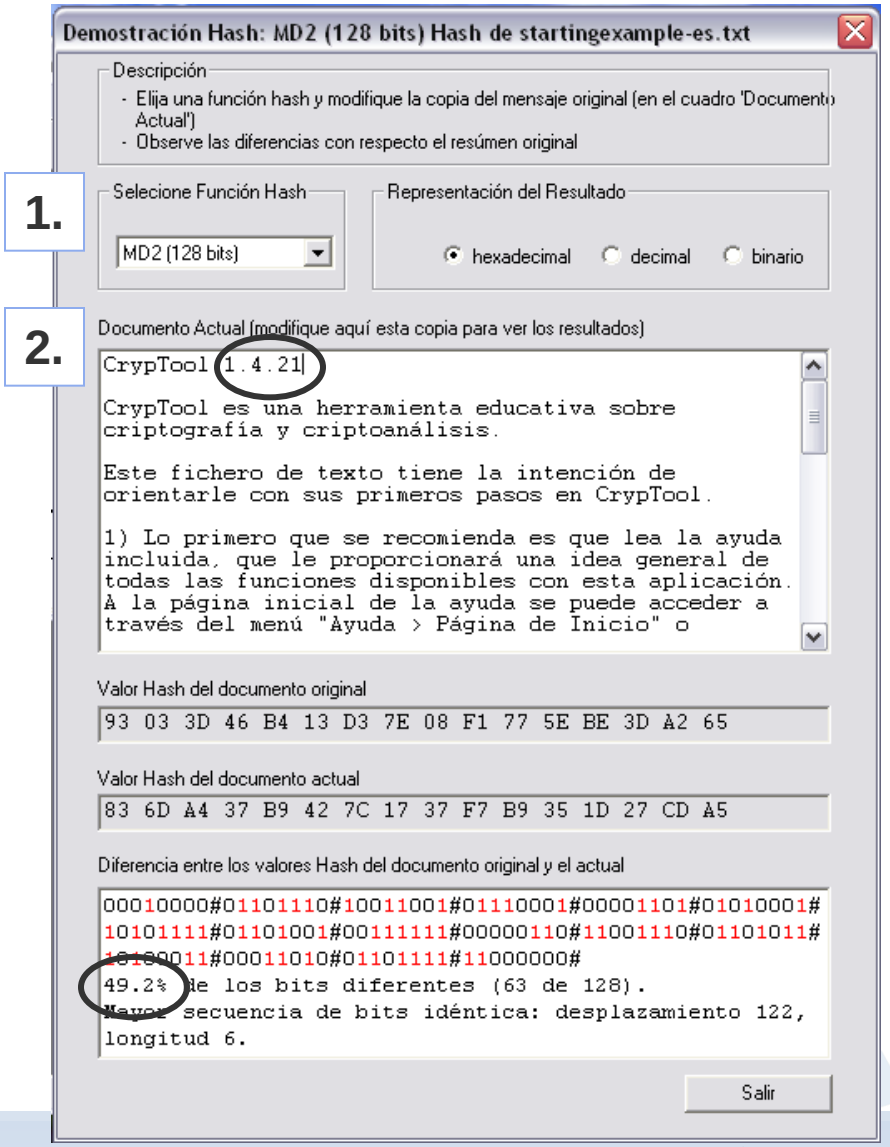
1. Seleccione una función hash
2. Modificar los caracteres del texto claro

Ejemplo:

Introduciendo un espacio después de “CrypTool” en el texto de ejemplo implica un cambio en 49,2% de los bits del valor hash generado.

Una buena función hash debe reaccionar sensiblemente frente a los más pequeños cambios en el texto claro – “efecto avalancha” (cambio pequeño, gran impacto).

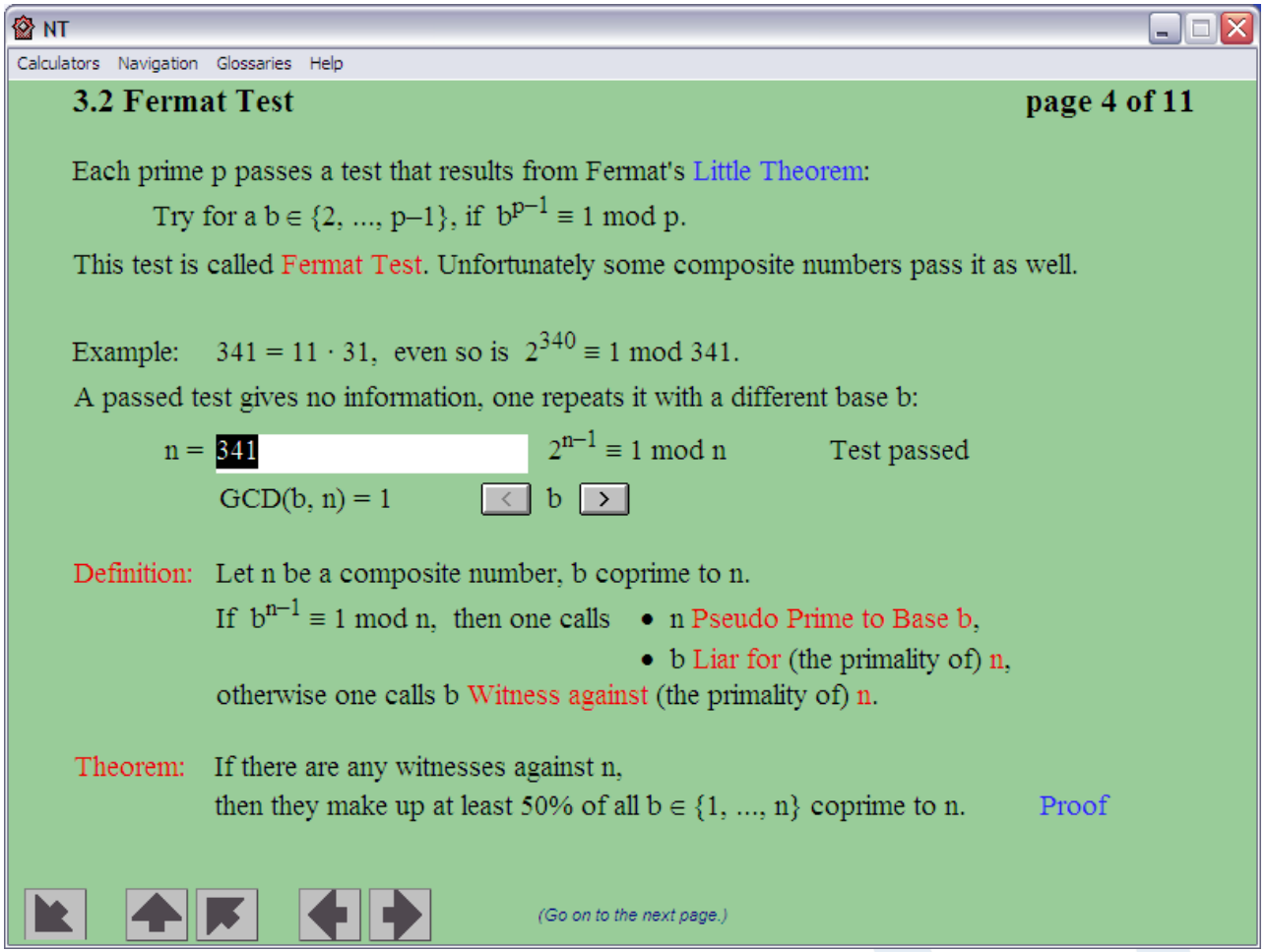
Menú: “Procedimientos Indiv.” / “Hash” / “Demostración Hash”



Ejemplos (20)

Herramienta para el aprendizaje de teoría de números

- **Teoría de Números** soportada por elementos gráficos y herramientas para probar
- **Temas:**
  1. Enteros
  2. Clases de Restos
  3. Generación de primos
  4. Criptografía de clave Pública
  5. Factorización
  6. Logaritmo Discreto



Menú: „Procedimientos Indiv." \ „Teoría de Números - Interactiva" \ „Herramienta para el aprendizaje sobre teoría de números"

Ejemplos (21)

Suma de puntos en curvas elípticas

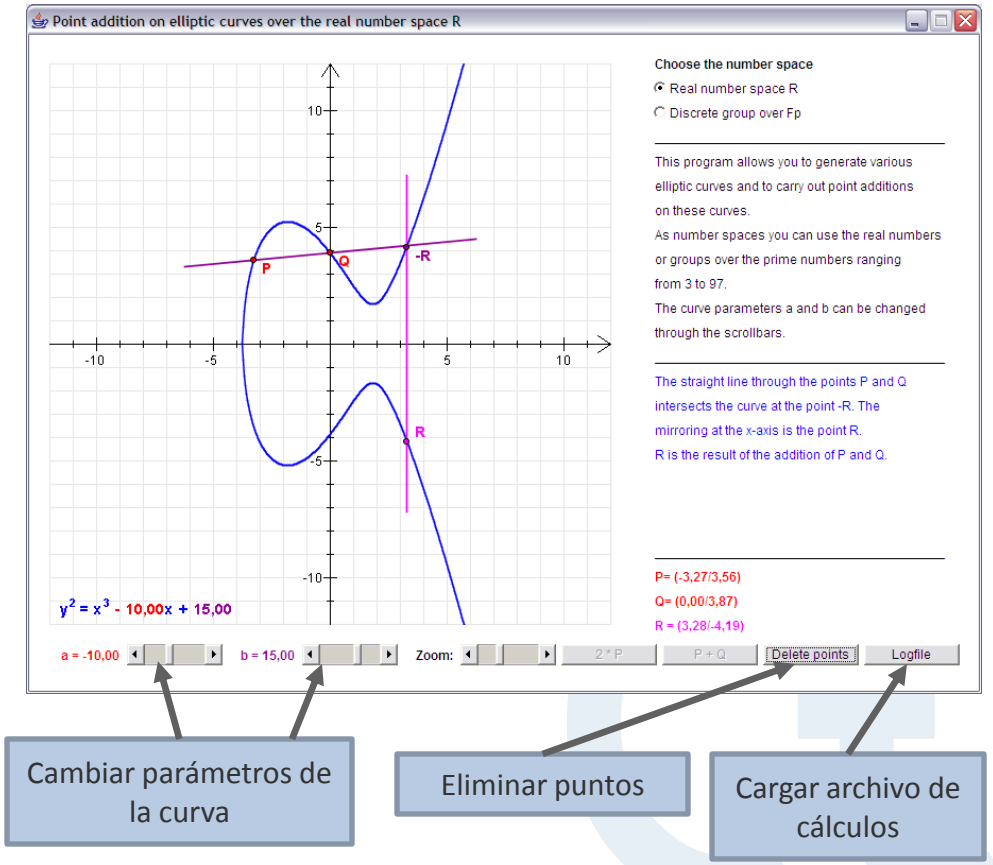
- Visualización de suma de puntos en curvas elípticas
- Bases para la criptografía basada en curvas Elípticas (ECC)

Ejemplo 1

- Marcar un punto P en la curva
- Marcar un punto Q en la curva
- Presionar el botón “P+Q”:
  - La línea recta que une los puntos P y Q e interseca a la curva en el punto –R
  - Reflejando en el eje de las X el resultado está en el punto R

Ejemplo 2

- Marcar el punto P en la curva
- Presionar el botón “2*P”:
  - La tangente al punto P que interseca a la curva en el punto -R
  - Reflejando en el eje X el resultado está en el punto R



Menú: “Procedimientos Indiv.” \ “Teoría de Números – Interactiva” \ “Suma de Puntos en Curvas Elípticas”

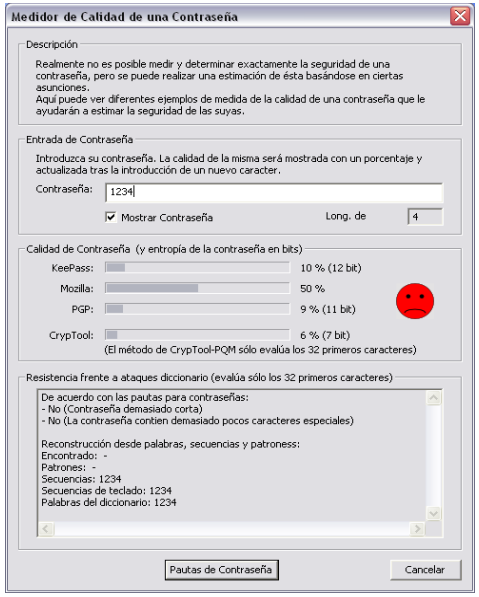
Ejemplos (22)

Medidor de Calidad de Contraseñas (Password Quality Meter “PQM”) 1

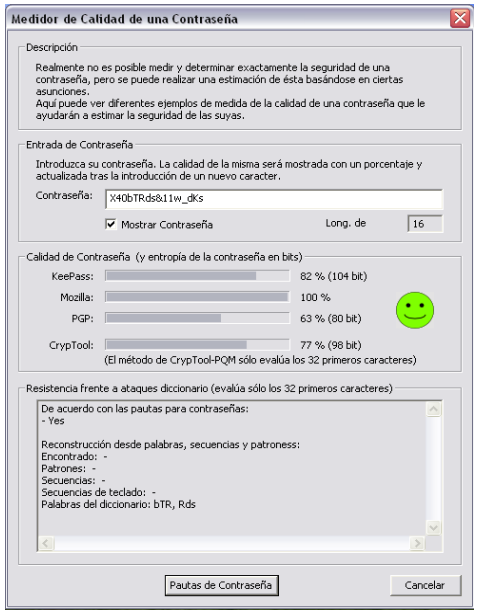
Funciones

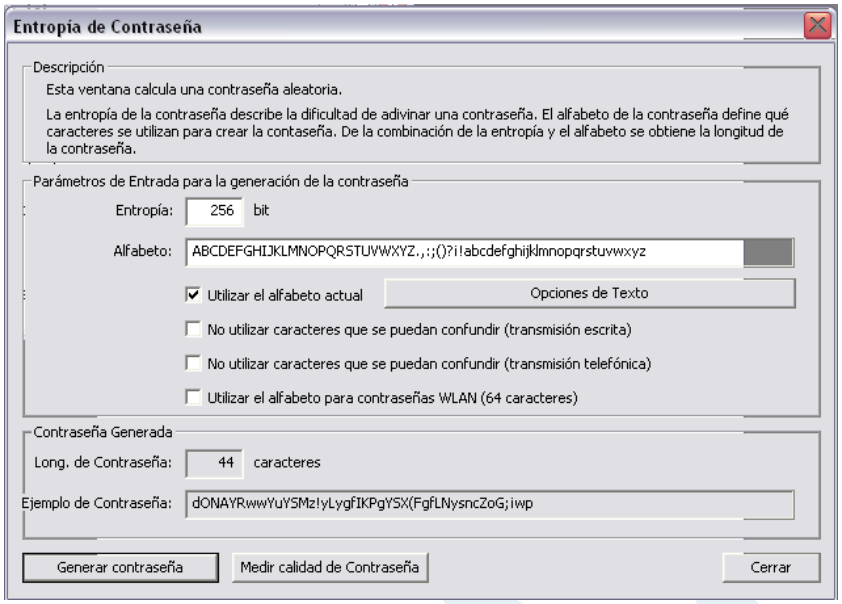
- Medida de la calidad de contraseñas
- Comparar con PQMs en otras aplicaciones: KeePass, Mozilla y PGP
- Medida experimental con el algoritmo de CrypTool
- Ejemplo: Entrada de una contraseña (mientras se muestra la contraseña)

Password: ***1234***



Password: ***X40bTRds&11w_dks***





Menú: “Procedimientos Indiv.“ \ “Herramientas“ \ “Medidor de Calidad de Contraseñas “

Menú: “Procedimientos Indiv.“ \ “Herramientas“ \ “Entropía de Contraseña“

Ejemplos (22)

Medidor de Calidad de Contraseñas (Password Quality Meter “PQM”) 2

Conclusiones del Medidor de Calidad de Contraseñas

- La calidad de la contraseña depende principalmente de la **longitud de la contraseña**.
- Se puede alcanzar una mayor calidad en la contraseña utilizando **distintos tipos de caracteres:** mayúsculas/minúsculas, números y caracteres especiales (**espacio de contraseña)**
- **Entropía de Contraseña** como indicador de la aleatoriedad de los caracteres de la contraseña o del espacio de contraseña (una mayor entropía aparece en una calidad de contraseña mejorada)
- Las contraseñas **NO deben existir en un diccionario** (nota: una comprobación con diccionario aún no se ha implementado en CrypTool).

Calidad de una contraseña desde la perspectiva de un atacante

- Ataque a una contraseña (con número de intentos ilimitado):
  1. **Ataque diccionario** clásico
  2. Ataque diccionario **con variantes** (p.ej. Combinaciones con números de 4 cifras: Verano2007)
  3. **Ataque por fuerza bruta** probando todas las combinaciones posibles (con parámetros adicionales como limitaciones en los tipos de conjuntos de caracteres)

⇨ Una buena contraseña se debe elegir para que los ataques 1. y 2. no la comprometan. Con respecto a los ataques de fuerza bruta, son importantes la longitud de la contraseña (al menos 8 caracteres) así como los conjuntos de caracteres utilizados.

Ejemplos (23)

Análisis por Fuerza Bruta 1

Análisis por fuerza bruta

Análisis por fuerza bruta optimizado bajo la suposición de que la clave se conoce parcialmente.

Ejemplo – Análisis con DES (ECB)

Intento de encontrar el resto de la clave para descifrar el texto cifrado (Suposición: el texto claro es un bloque de 8 caracteres ASCII)

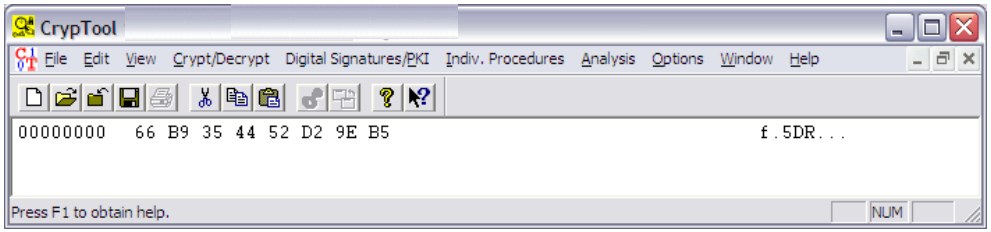
Clave (Hex)	Texto Cifrado (Hex)
68ac78dd40bbefd*	66b9354452d29eb5
0123456789ab****	1f0dd05d8ed51583
98765432106*****	bcf9ebd1979ead6a
0000000000******	8cf42d40e004a1d4
000000000000****	0ed33fed7f46c585
abacadaba*******	d6d8641bc4fb2478
ddddddddddd******	a2e66d852e175f5c

Ejemplos (23)

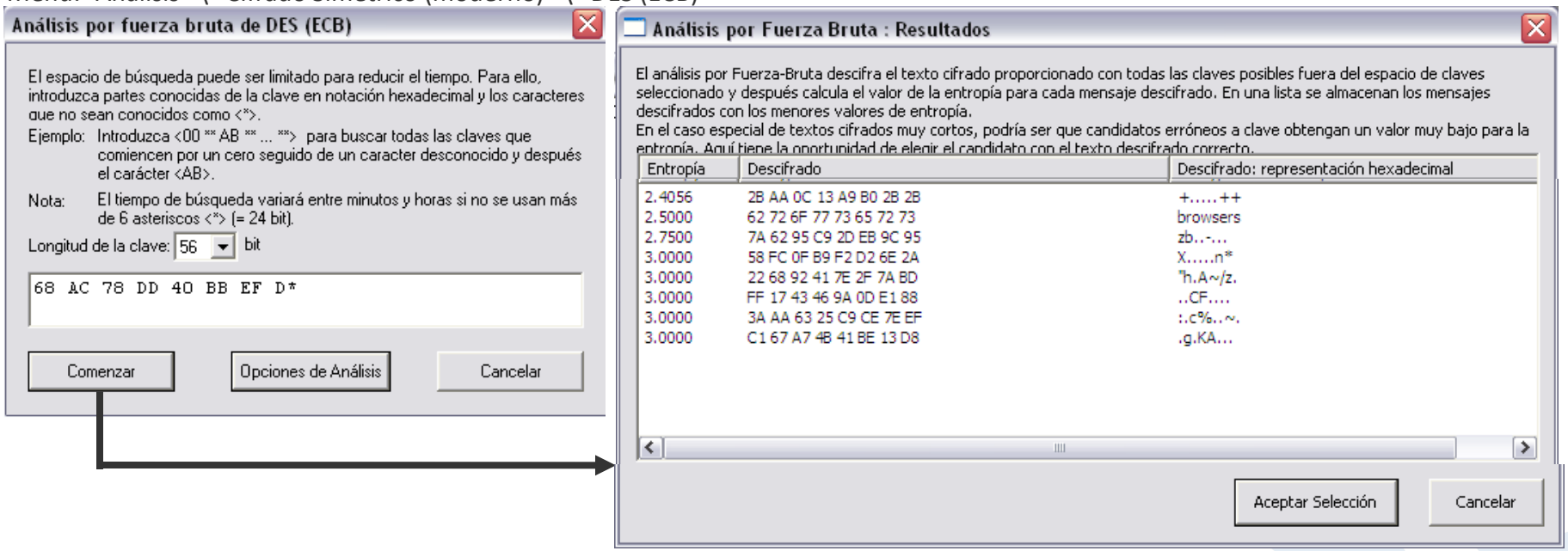
Análisis por Fuerza Bruta 2

1. Entrada de texto cifrado
2. Utilizar análisis por fuerza bruta
3. Introducir parte de la clave conocida
4. Empezar análisis por fuerza bruta
5. Análisis de los resultados: una baja entropía evidencia un posible descifrado. Sin embargo, a causa del texto claro corto utilizado en este ejmemplo, el resultado correcto no es el que tiene la entropía más baja.

Utiliza “Ver” \ “Mostrar como código hexadecimal”



Menú: “Análisis” \ “Cifrado Simétrico (moderno)” \ “DES (ECB)”



Ejemplos (24)

Escítala / Rail Fence

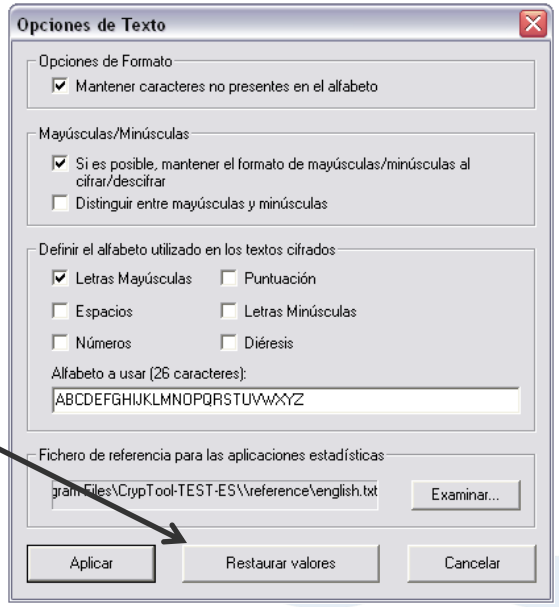
Escítala y Rail Fence

- Transposiciones mezclan el orden de las letras en el texto claro
- **Parámetro de Transposición**
  - Número de esquinas (Escítala)
  - Número de líneas (Rail Fence)
  - Offset

Menú: “Cifrado/Descifrado” \ “Simétrico (clásico)” \ “Escítala/Rail Fence …”

Opciones de Texto

- Opciones generales de texto (Menú: “Opciones” \ “Opciones de Texto…”)
- Opciones de formato para texto claro y cifrado
- Distinción entre mayúsculas y minúsculas
- Alfabeto para el procesamiento de texto (muestra los caracteres que deben ser cifrados/descifrados)
- Vuelva a establecer los valores por defecto a través del botón “Restaurar valores”
- Cree los patrones de referencia estática en forma dinámica.



Ejemplos (25)

Cifrado Hill / Análisis Hill (1)

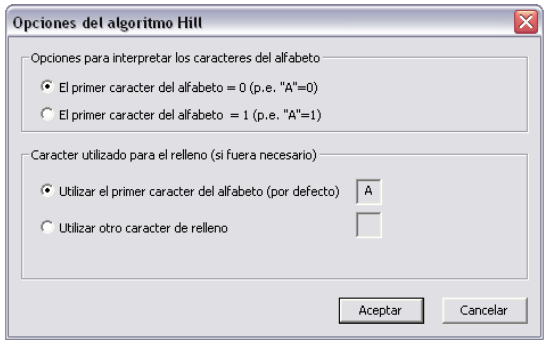
Cifrado Hill

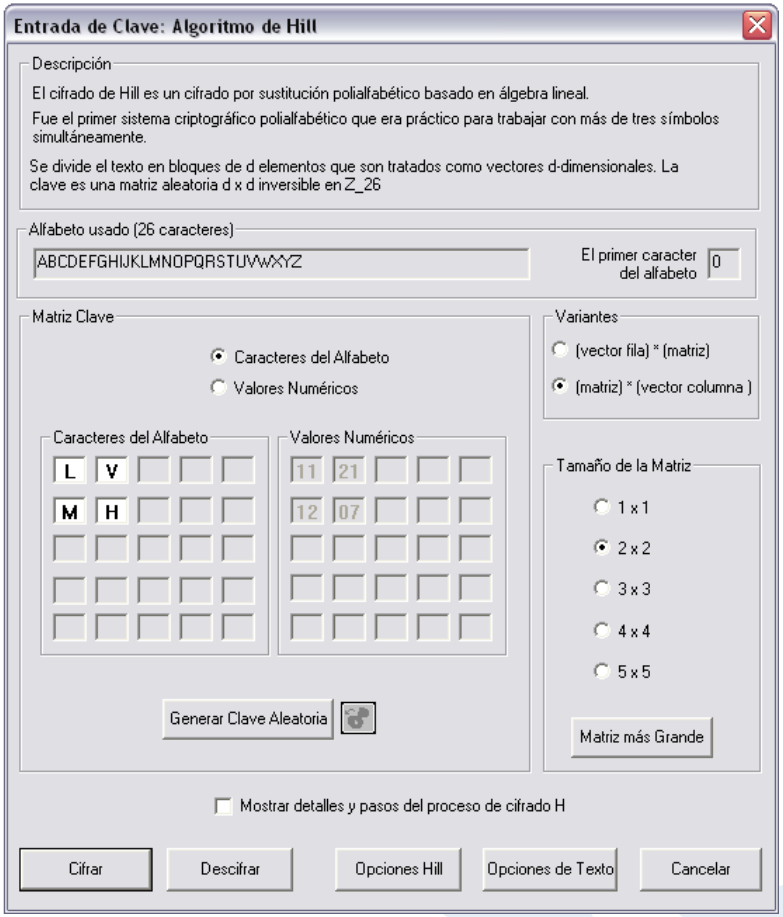
- Cifrado de sustitución poligráfico
- Basado en álgebra lineal

Clave

- Caracteres de alfabeto (Ver opciones de texto) o valores numéricos
- Ingresar clave o generar una aleatoria
- Seleccionar parámetro de multiplicación
- Tamaño de la matriz
- Opciones

Menú:
"Cifrar/Descifrar" \
"Simétrico (clásico)" \
"Hill …"





Ejemplos (25)

Cifrado Hill / Análisis Hill (2)

Cifrado Hill

- Texto de ejemplo con la clave: `LVMH`

Análisis Hill (texto claro conocido)

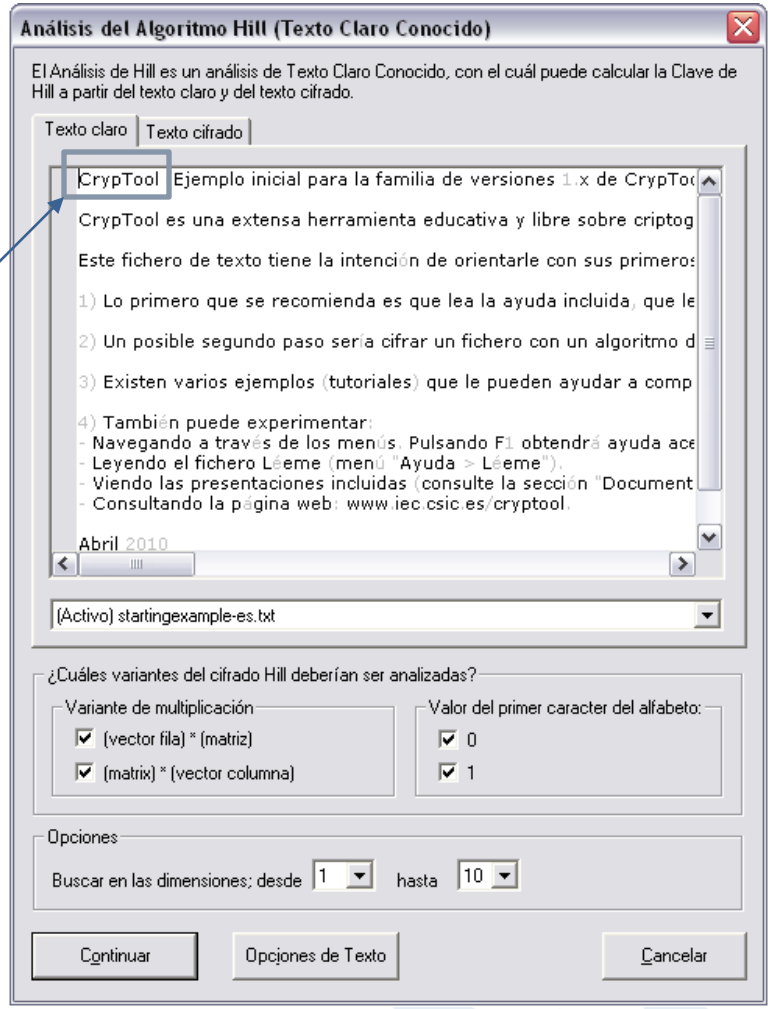
1. **Texto claro / texto cifrado - Largo**

- Seleccionar texto plano (startingexample-es.txt)
- Seleccionar texto cifrado
  (Cifrado Hill de <startingexample-es.txt>)
- “Continuar” para buscar la clave

2. **Texto claro / texto cifrado - Reducido**

- Eliminar todo, excepto el inicio del texto claro (“CrypTool”)
- Reducir texto cifrado a “PnhdJovl”
- “Continuar” encuentra la clave correcta

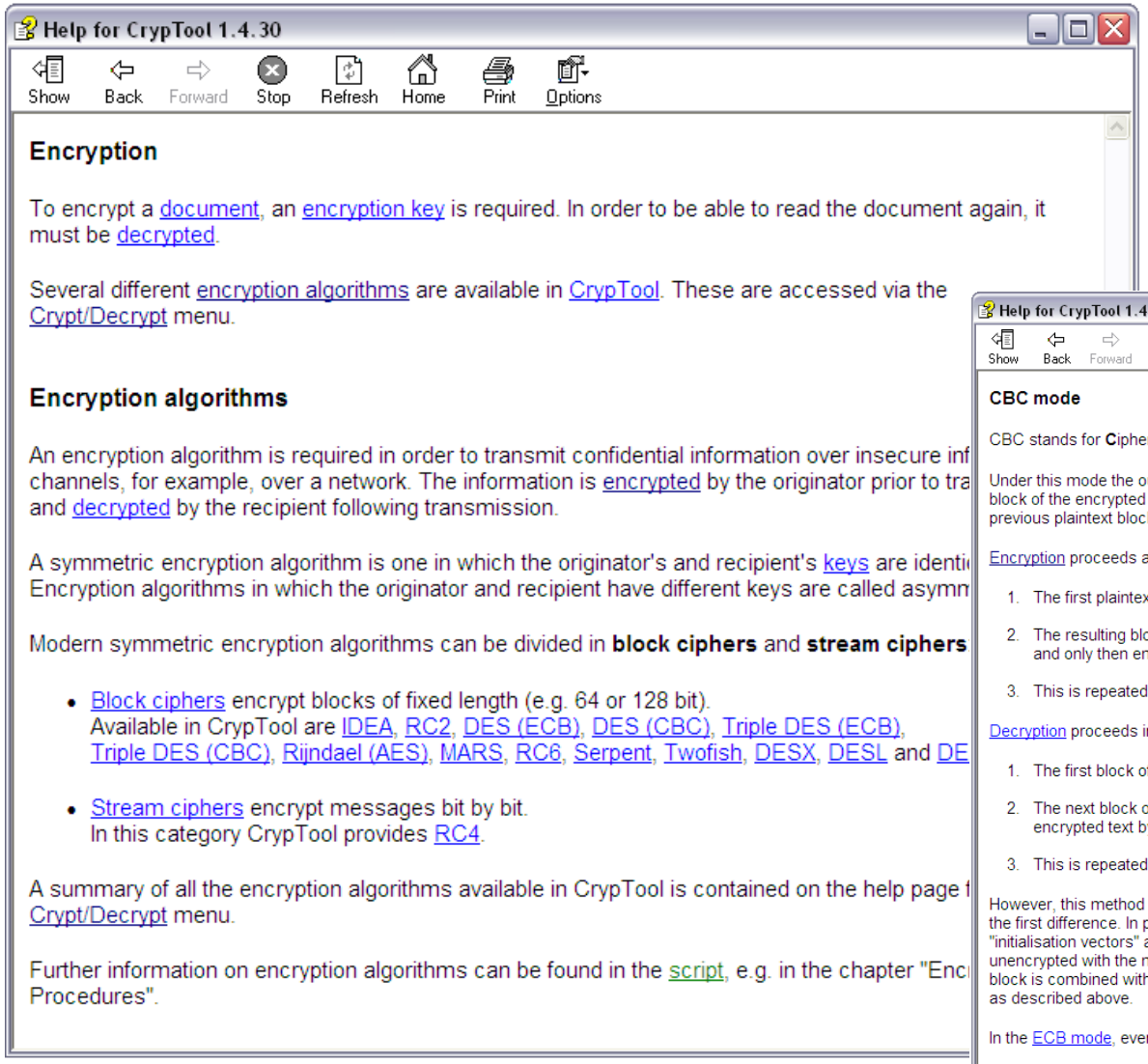
¿Qué cantidad de texto claro/cifrado es necesario para encontrar la clave de cifrado correcta?

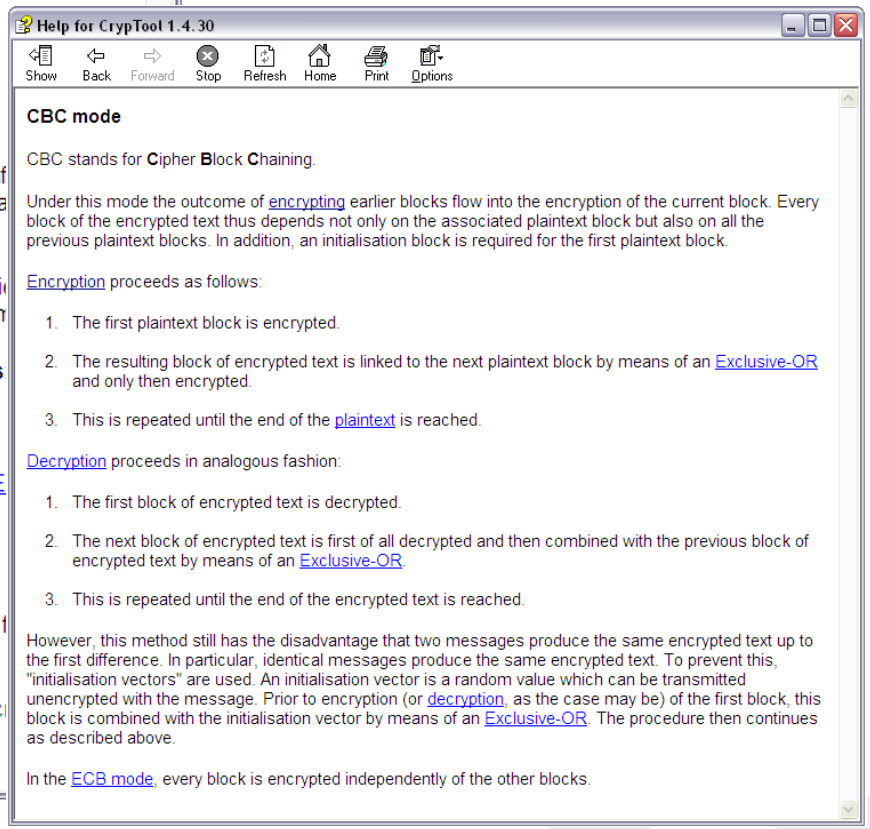


Menú: “Análisis” \ “Simétrico (clásico)” \ “Texto claro conocido” \ “Hill…”

Ejemplos (26)

Ayuda Online de CrypTool (1)





Menú: “Ayuda” \ “Página de Inicio”

Ejemplos (26)

Ayuda Online de CrypTool (2)

Help for CrypTool 1.4.30

Hide | Back | Forward | Stop | Refresh | Home | Print | Options

Contents | Index | Search

Type in the keyword to find:

lattice reduction

- Lattice reduction
- Liability (exclusion)
- License terms
- Line wrap
- Links
- Literature
- MARS encryption algorithm
- MD2 hash value
- MD4 hash value
- MD5 hash value
- Menu (overview of all menus)
- Miracl
- Modular transformation
- Modulo operator
- Monoalphabetic substitution encryp
- Network authentication
- N-gram
- Nihilist encryption algorithm
- NIST
- Normal distribution
- NSA
- NTL
- Number Shark
- Number system
- Number theory
- Offset
- One-time pad
- OpenGL
- OpenPGP
- OpenSSL
- Options
- Overview / Subsumption / Broader C
- Padding
- Parent window
- Password
- Pattern search

Display

Menu Lattice Based Attacks on RSA (Menu Individual Procedures \ RSA Cryptosystem)

The menu **Lattice Based Attacks on RSA** contains the following commands:

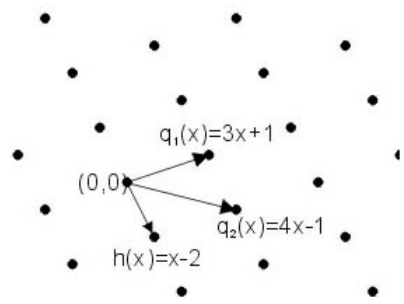
Factoring with a Hint	Attacks RSA with lattice reduction algorithms, if a part of one of the primes of N is known.
Attack on Stereotyped Messages	Attacks RSA with lattice reduction algorithms, if a part of the original cleartext of an intercepted ciphertext is known and if e is small.
Attack on Small Secret Keys	Attacks RSA with lattice reduction algorithms, if d is too small compared to N.

All attacks presented here are based on a common approach: first the task of breaking RSA is transformed into finding the root of a polynomial modulo an integer (mostly **N**) but to find such a root is a difficult problem.

To solve this problem further polynomials are generated which are known to have the same root. From the coefficients of these polynomials a latticebase is built. This is then reduced with, i.e. the LLL-algorithm to find a small vector.

From this newly found short vector a new polynomial is built. It can be proven that if the vector is short enough, the polynomial has the desired root not only modulo **N**, but also over the integers.

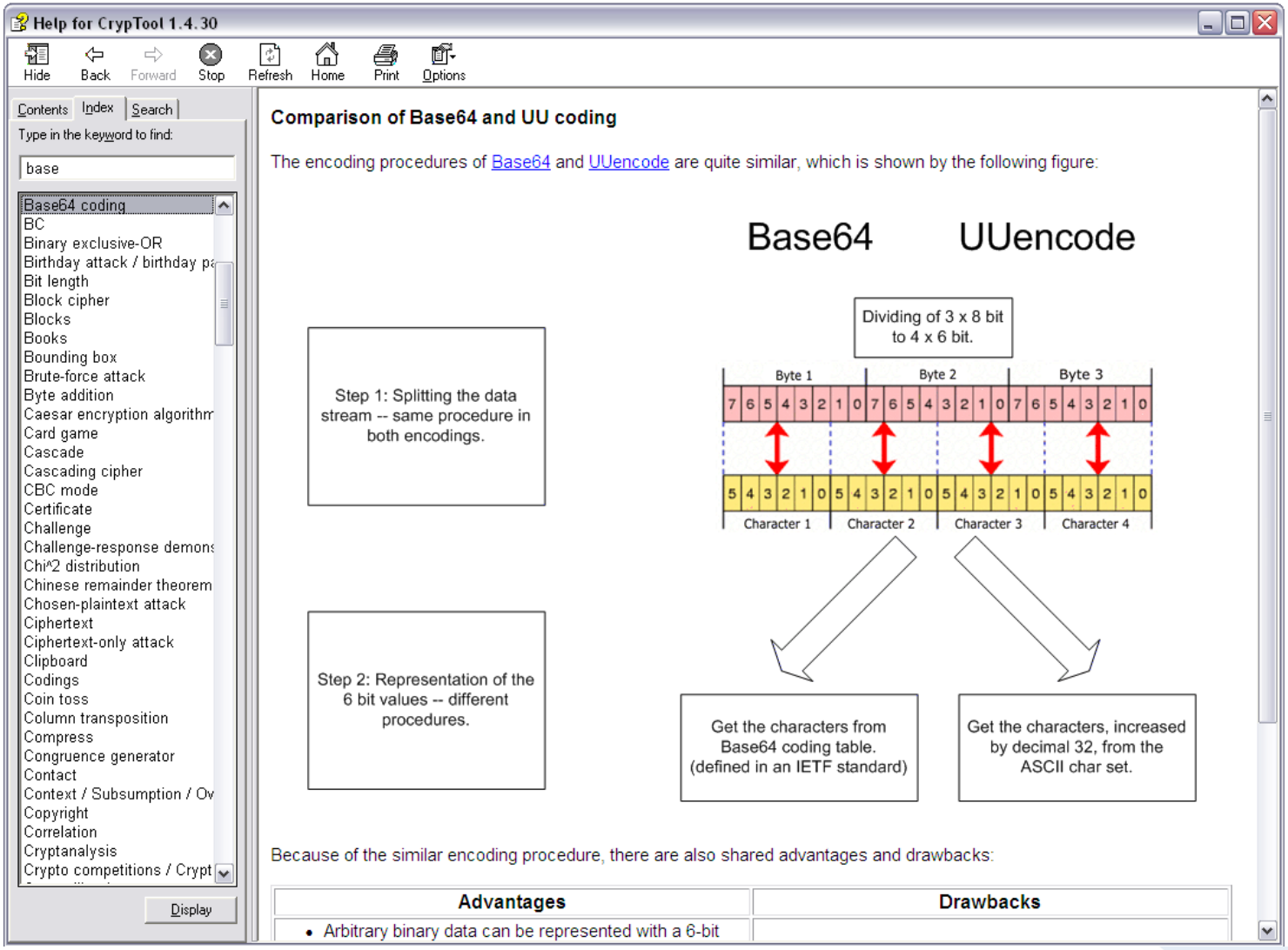
Example:



The polynomial $q_1(x) = 3x+1$ has a root x_0 **modulo 7**. It is supposed, that the polynomial $q_2(x) = 4x-1$ has the same root x_0 **modulo 7**. From these polynomials the vectors b_1=[3 1] and b_2=[4 -1] are built. All integer linear combinations of these vectors form points in a lattice. The Figure on the left shows a part of this lattice. Each point of the lattice now can again be interpreted as a polynomial having the desired root. A short vector of the lattice is b_3=[1 -2] from which the polynomial $h(x) = x-2$ is built. this polynomial has a root in x_0=2 over the integers as well als **modulo** 7. That x_0=2 is also a root of the polynomials $q_1(x)$ and $q_2(x)$ **modulo** 7 can be easily established.
($3x_0+1=7$, 7 **modulo** 7 = 0)

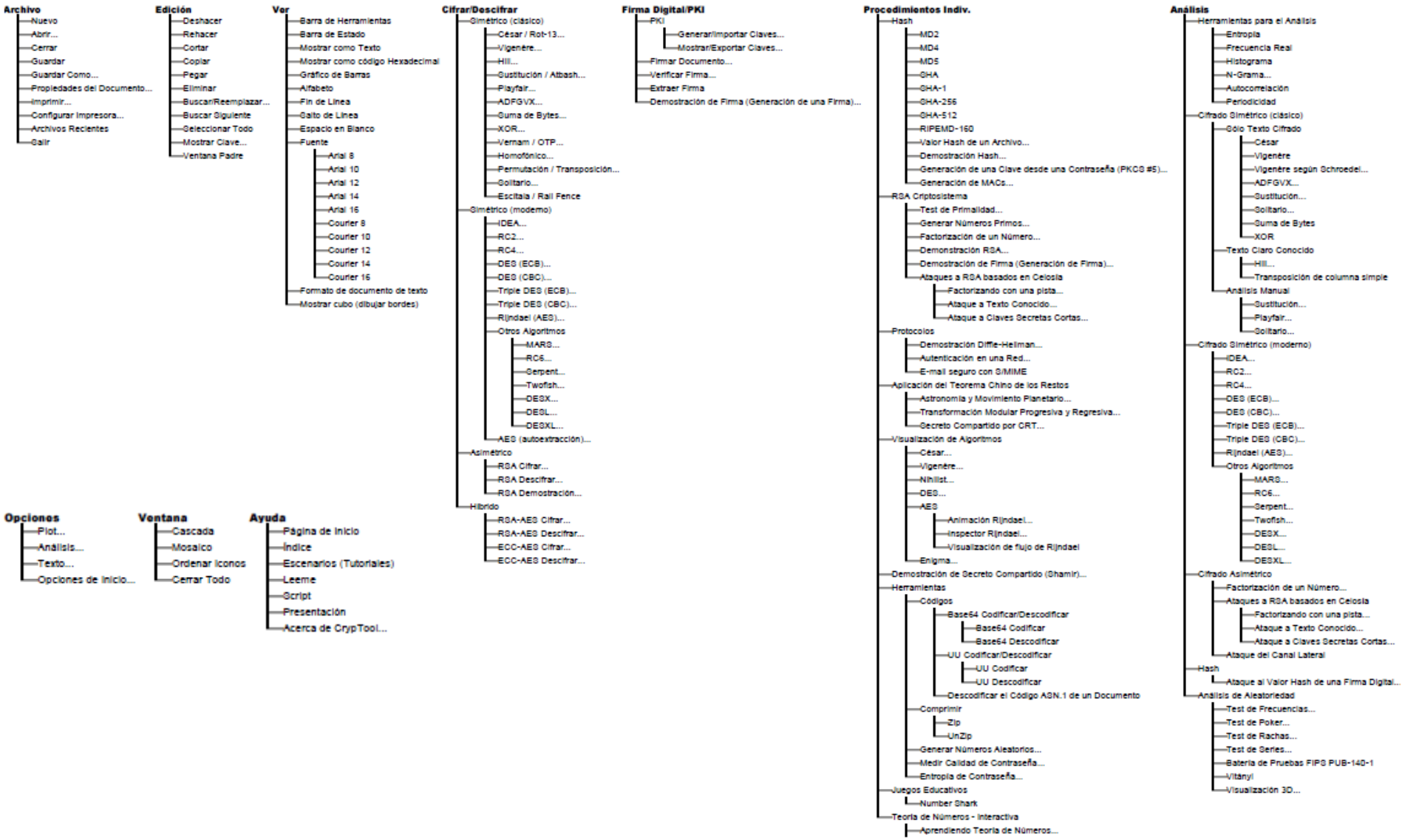
Ejemplos (26)

Ayuda Online de CrypTool (3)



Ejemplos (26)

Vista de árbol de menúes de CrypTool 1.4.30



Contenido

Desarrollo de CrypTool en el Futuro (1)

Plan después de la publicación de 1.4.30 (ver archivo Léeme)

CT1	Test FIPS para investigar partes de tamaño mayor a 2500 bytes
JCT	Acuerdos de clave tri-partita
JCT	Visualización de la interoperabilidad de los formatos S/MIME y OpenPGP
JCT	Análisis de entropía
JCT	Grille, Vigenère Autoclave, Criptoanálisis interactivo de Cifrados clásicos
JCT	Cifrados de análisis de transposición usando el algoritmo ACO
JCT	Visualización de las pruebas de cero conocimiento
JCT	Visualización del acuerdo de clave Quantum, Protocolo BB84
JCT	Visualización del ataque SETUP contra la generación de claves RSA (Kleptografía)
JCT	Action-History con característica adicional para crear y reproducir cualquier cifrado (cascada)
CT2	Visualización comprensible del tema de los números primos
CT2	GNFS (Tamiz general del campo del número)
CT2	Cifrado y criptoanálisis automatizado de la máquina Enigma y tal vez de Sigaba
CT2	Ataque del Cubo (I. Dinur y A. Shamir, "Cube Attacks on Tweakable Black Box Polynomials“, 2008)
CT2	Demonstración de la falsificación de la firma RSA de Bleichenbacher
CT2	Demonstración virtual de números de tarjetas de crédito (enfoque contra del abuso en tarjetas de crédito)
CT2	Cifrado WEP y análisis WEP
CT2	Búsqueda masiva de patrones
CT2	Framework para criptoanálisis distribuido
CT2	Demonstración de seguridad de SOA (mensajes SOAP a través de seguridad WS entre los participantes)
CT2	Framework para crear y analizar cifrados de flujo LFSR
CT2/JCT	Creación de una versión en línea de comandos para un procesado por lotes
CT2/JCT	Moderna arquitectura *pure* plugin con plugins cargados
Todo	Parametrización adicional / Incrementando la flexibilidad de los algoritmos presentes
Ideas	Visualización del protocolo SSL // Demonstración de criptografía visual

CT1 = CrypTool 1.x

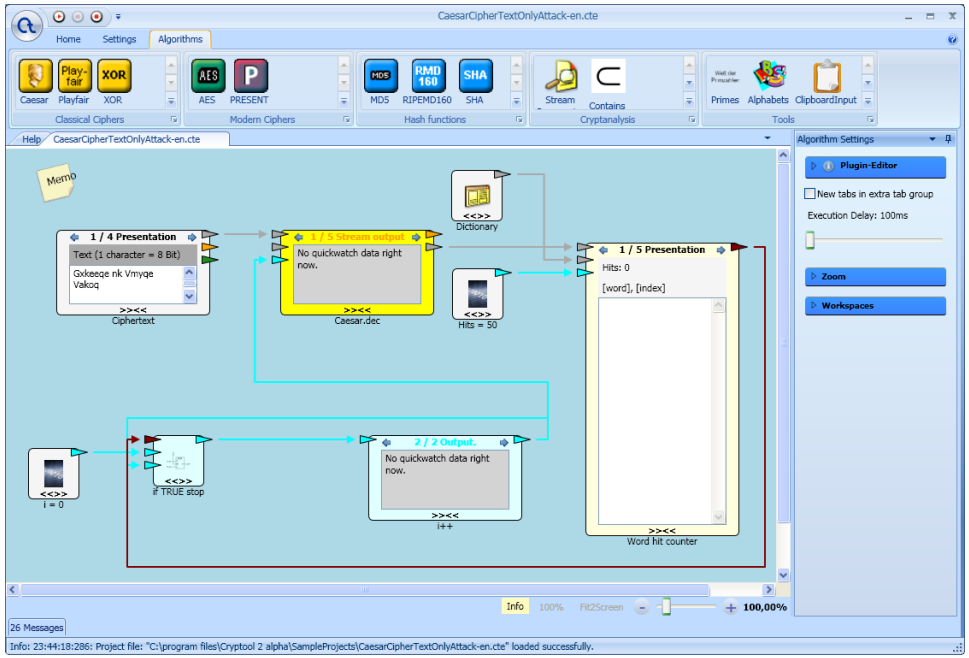
Nuevas versiones:

CT2 = CrypTool 2.0
JCT = JCrypTool
(Ambos presentados a continuación)

Desarrollo de CrypTool en el Futuro (2)

En desarrollo: Las dos versiones sucesoras de CT v1 (ver archivo Léeme)

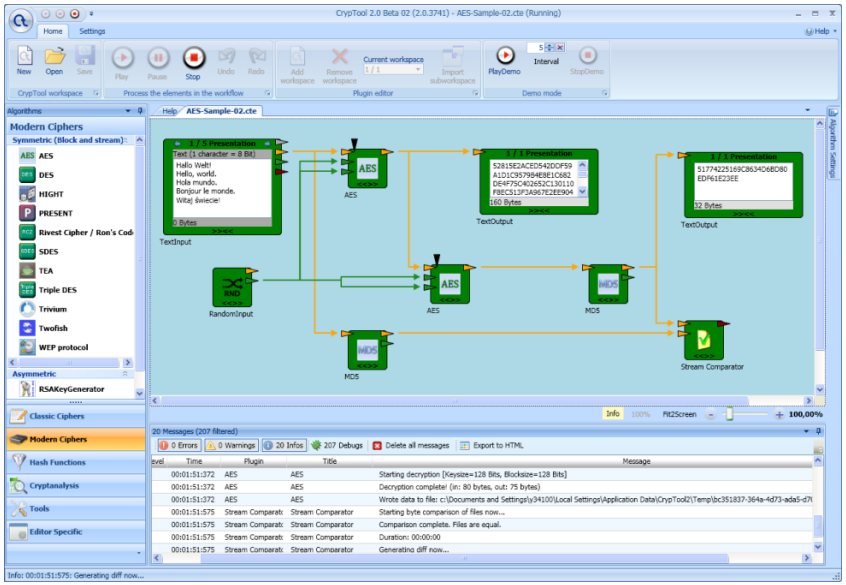
1. JCT: Portabilidad y rediseño de CrypTool en Java / SWT / Eclipse 3.6 / RPC
   - ver: http://jcryptool.sourceforge.net
   - Release Candidate RC3 está disponible para desarrolladores y usuarios (Julio 2010)
2. CT2: Portabilidad y rediseño de la versión en C++ con C# / WPF / VS2010 / .NET 4.0
   - Sucesor directo de las versiones actuales: perite programación visual, etc.
   - Descargar de: http://cryptool2.vs.uni-due.de/index.php?page=14&lm=1&ql=4
   - La versión Beta3 está disponible desde Agosto 2010 (Actualizada continuamente desde Junio de 2008)



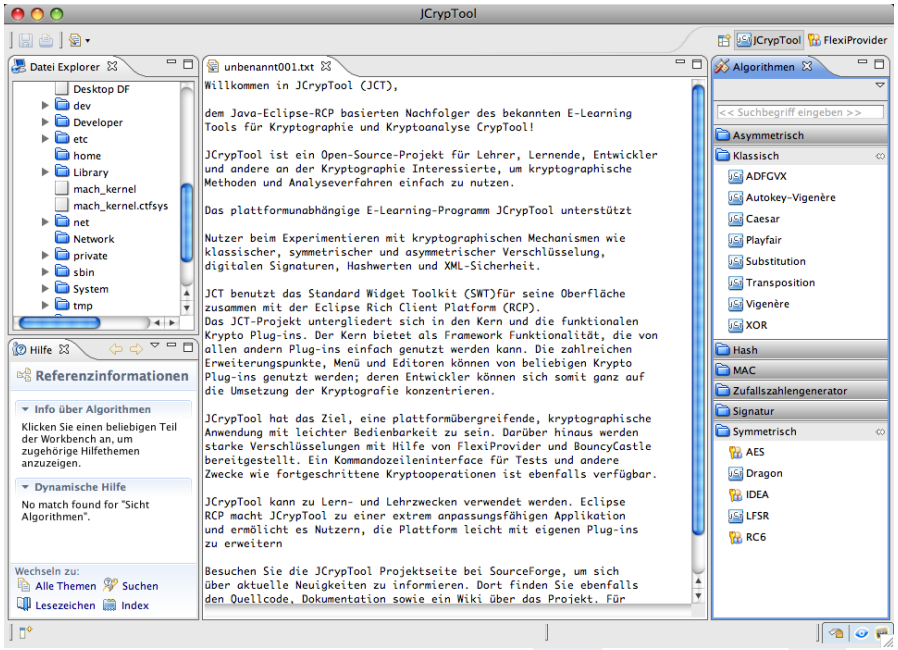
CrypTool 2 (CT2)

JCrypTool (JCT)

Desarrollo de CrypTool en el Futuro (3)



CrypTool 2 (CT2)



JCrypTool (JCT)

CrypTool como un “Framework”

Propuesta

- Reutilizar el amplio conjunto de algoritmos, incluyendo las librerías y los elementos de la interfaz como base
- Entrenamiento gratuito en Frankfurt, cómo empezar con el desarrollo de CrypTool
- Ventaja: Tu propio código no “desaparece”, se mantendrá

Entorno de desarrollo actual para CT1: **Microsoft Visual Studio C++ , Perl, Subversion Source-Code Management**

- Hasta CrypTool 1.4.30: Visual C++ .NET (= VC++ 9.0)(= Visual Studio 2008 Standard)
- Descripción para desarrolladores: ver readme-source.txt
- Descarga: de fuentes y binarios de las publicaciones. Para obtener los archivos fuente de las betas actuales, por favor vea el repositorio de subversiones.

Entornos de desarrollo para CT2 y JCT

- CT2 – versión C# : .NET con Visual Studio 2010 Express Edition (gratis) y WPF
- Java – versión Java: Eclipse 3.6, RCP, SWT (gratis)

CrypTool – Petición de Colaboración

Toda colaboración con el proyecto se agradece enormemente

- Realimentación de información, críticas, sugerencias e ideas
- Integración de algoritmos adicionales, protocolos, análisis (consistencia y completitud)
- Desarrollo de asistencia (programación, diseño, traducción, prueba)
  - Para el proyecto C/C++ actual
  - Para los nuevos proyectos (preferencialmente)
    - Proyecto C# : "CrypTool 2.0" = CT2
    - Proyecto Java : "JCrypTool" = JCT
  - Especialmente se invita al desarrollo adicional a las Universidades que utilizan CrypTool para propósitos educativos.
- Ejemplos de tareas abiertas se encuentran en las páginas de desarrollo respectivas:
  - CT2: Ver la lista: http://cryptool2.vs.uni-due.de, voluntarios, tareas actuales
  - JCT: Ver: wiki http://sourceforge.net/apps/mediawiki/jcryptool/index.php?title=CurrentDevelopment
- Las colaboraciones significativas se pueden referenciar por nombre (en la ayuda, Léeme, ventana Acerca de, o en la página web de CrypTool).
- Actualmente CrypTool posee más de 6000 descargas al mes (de las cuales un poco más del 50% se realiza sobre la versión en inglés).
- Las versiones Betas de las dos herramientas sucesoras (JCT y CT2) registran ya más de 1000 descargas mensuales.

CrypTool – Resumen

- EL programa de aprendizaje electrónico para criptología
- Un proyecto de Código Abierto con más de 10 años de éxito
- Más de 400.000 descargas
- Uso internacional en escuelas, universidades, así como empresas y agencias del gobierno
- Amplia ayuda online y documentación
- Disponible gratuitamente y con soporte multi-idioma

Contacto

Prof. Bernhard Esslinger

Universidad de Siegen
Facultad 5, Computación y Sistemas de Información

Deutsche Bank S.A.
Director, IT Security Manager

esslinger@fb5.uni-siegen.de

www.cryptool.org

www.cryptool.com

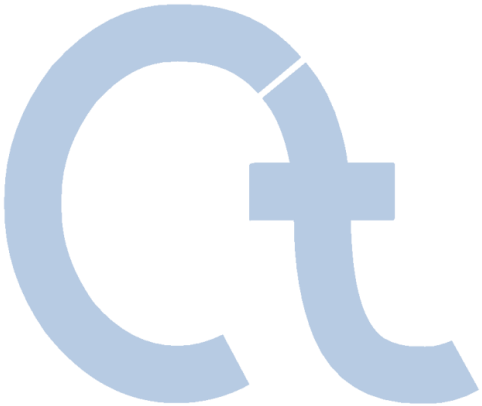
www.cryptool.de

www.cryptool.es

www.cryptool.pl

Contactos adicionales: ver Léeme en la carpeta CrypTool

Contenido



I. CrypTool y Criptología – Visión General

II. Características de CrypTool

III. Ejemplos

IV. Proyecto / Perspectiva / Contacto

Apéndice

(Bibliografía adicional, sitios relacionados, descargas, etc.)

Bibliografía Adicional

Para introducirse en la Criptología

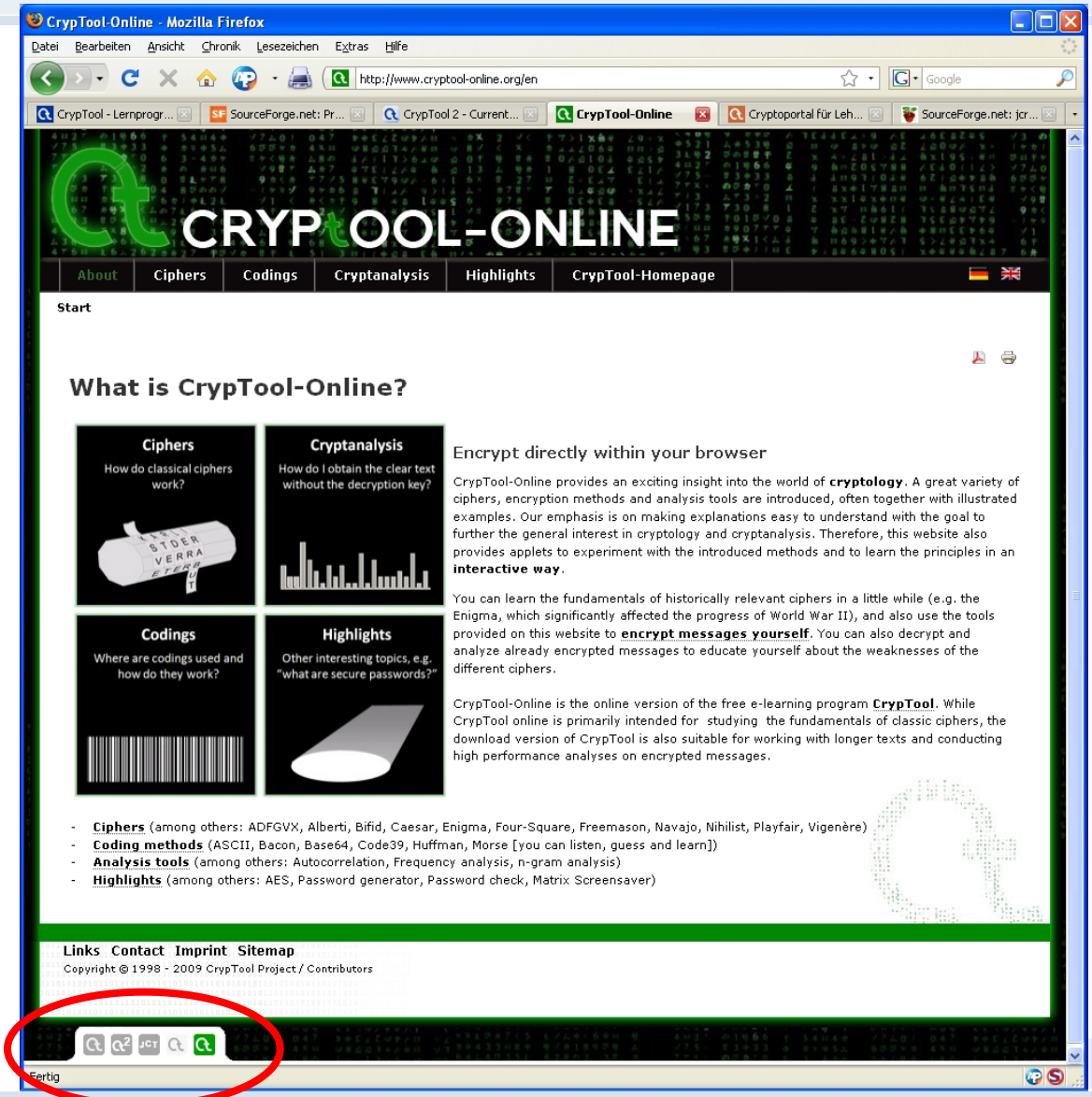
- **Simon Singh, *"Los códigos secretos"*, 2000, Doubleday**
- **J. Ortega y Miguel Ángel López Guerrero, *"Introducción a la Criptografía"*, 2006**
- **Jorge Ramió, "Libro Electrónico de Seguridad Informática y Criptografía"**
- **Johannes Buchmann, "*Introduction to Cryptography*", 2nd edition, 2004, Springer [inglés]**
- **Paar / Pelzl: *„Understanding Cryptography – A Textbook for Students and Practitioners "*, 2009, Springer [inglés]**
- **Klaus Schmeh, "*Codeknacker gegen Codemacher. Die faszinierende Geschichte der Verschlüsselung*", 2nd edition, 2007, W3L [alemán]**
- **[HAC] Menezes / van Oorschot / Vanstone, *"Handbook of Applied Cryptography"*, 1996, CRC Press**
- **van Oorschot / Wiener, *"Parallel Collision Search with Application to Hash Functions and Discrete Logarithms"*, 1994, ACM [inglés]**
- **Bibliografía adicional sobre criptografía – ver también la página web de CrypTool y la bibliografía de la ayuda online de CrypTool (p.ej. por Wätjen, Salomaa, Brands, Schneier, Shoup, Stamp/Low, …)**
- **La importancia de la criptografía en el amplio contexto de la seguridad en TI y la gestión de riesgos**
  - Ver p.ej. Kenneth C. Laudon / Jane P. Laudon / Detlef Schoder, "Wirtschaftsinformatik", 2005, Pearson, chapter 14 [alemán]
  - Ver Wikipedia (http://en.wikipedia.org/wiki/Risk_management) [inglés]
  - Página de CrypTool: http://cryptool.com/index.php/en/cryptool-for-awareness-aboutmenu-74.html

www.cryptool.org / .com / .de / .es / .pl

www.cryptool-online.org

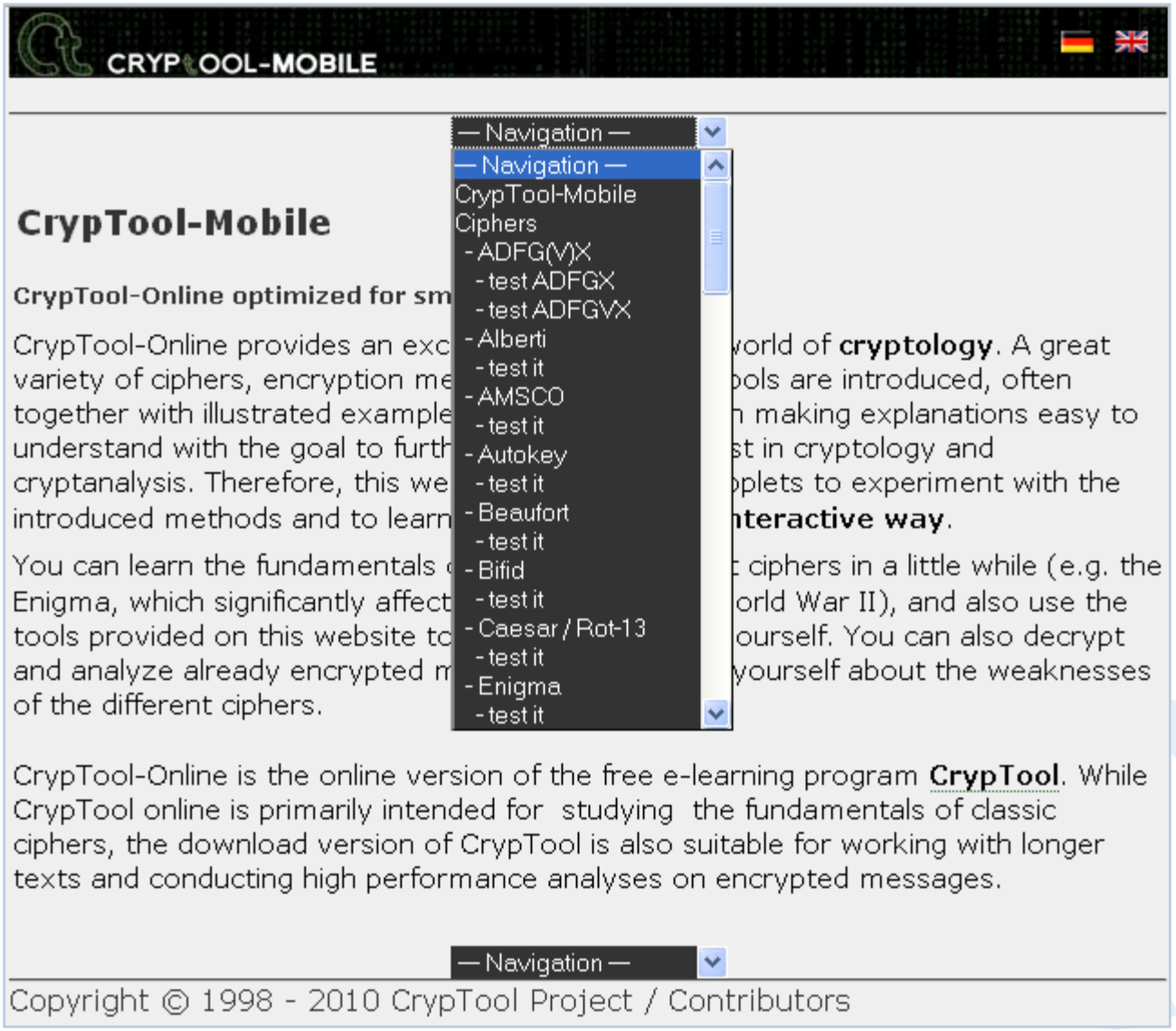
Los miembros de la familia de CrypTool.
Sitios relacionados:

- **CrypTool** (CT1)
- **CT2** – para desarrolladores
- **JCT** – para desarrolladores
- **CrypTool-Online + CrypTool-Mobil**
  Pruebe métodos criptográficos en su navegador y en su teléfono móvil.
- **CryptoPortal** para profesores (actualmente sólo en alemán)
- **Mystery Twister C3 (MTC3)**
  Concurso de desafíos criptográficos



m.cryptool.org

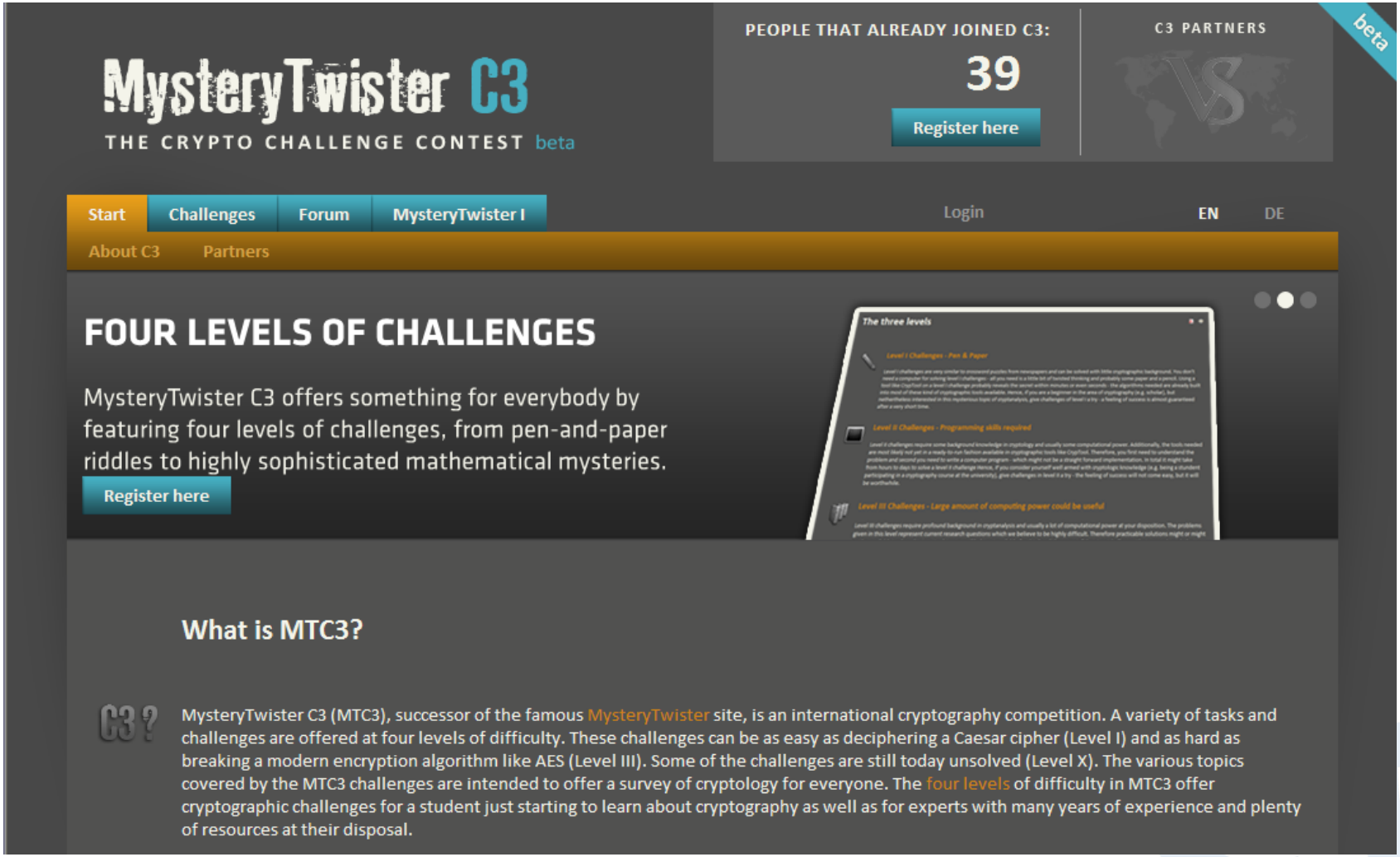
Viva la criptografía desde su Smartphone



www.cryptoportal.org

El portal para profesores se encuentra sólo en alemán. Se acepta gustosamente ayuda para su versión en español o en inglés.

http://www.mysterytwisterc3.org/



Mystery Twister C3 es un concurso de desafíos criptográficos.

¡Descargue el software y el CrypTool Script!