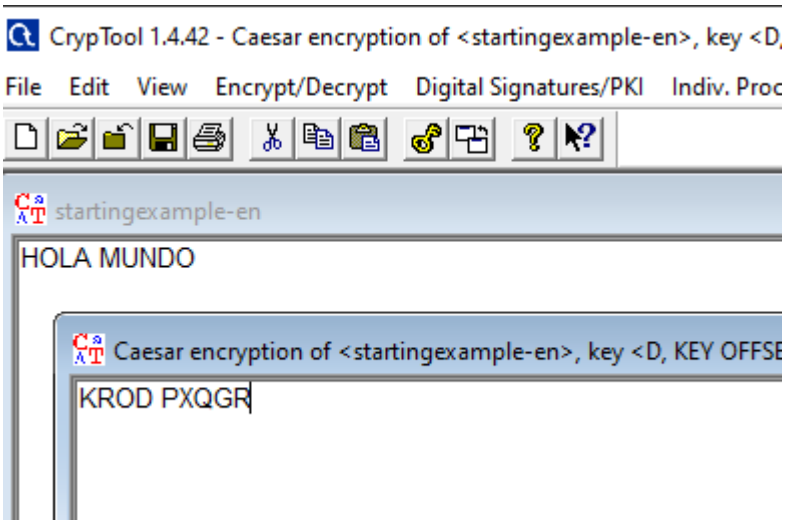
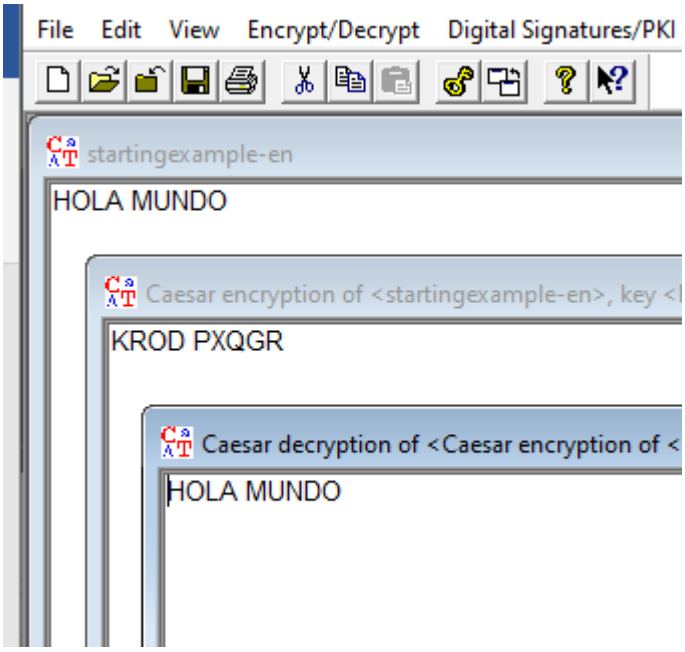
Práctica: Ejercicios básicos de Criptografía con CrypTool

🔷 1. Cifrado César básico

Objetivo: aplicar el cifrado de sustitución más sencillo.

1. Abre CrypTool y escribe en el editor:
2. HOLA MUNDO
3. Menú: **Encrypt / Decrypt → Symmetric (classic) → Caesar**.
4. Selecciona desplazamiento = **3**.
5. Anota el texto cifrado.
6. Usa la opción de descifrar para volver al texto original.



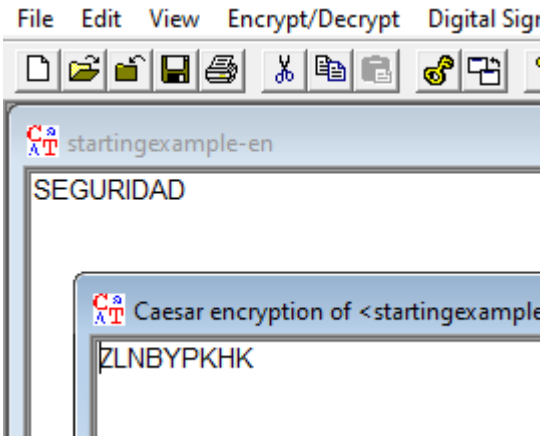


👉 **Resultado esperado:** comprobar cómo funciona el desplazamiento de letras.

🔷 2. César con otra clave

Objetivo: ver cómo cambia con distinta clave.

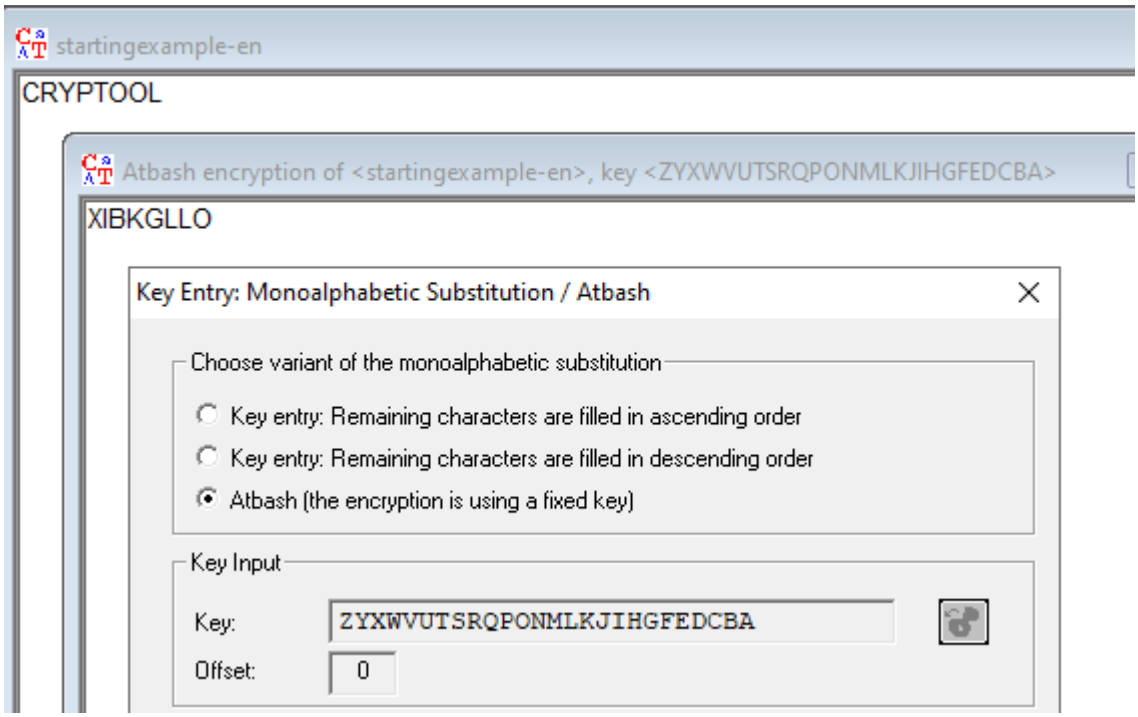
1. Escribe:
2. SEGURIDAD
3. Cifra con **César** usando desplazamiento = **7**.
4. Anota el resultado.



🔷 3. Sustitución monoalfabética

Objetivo: ver otro tipo de cifrado clásico.

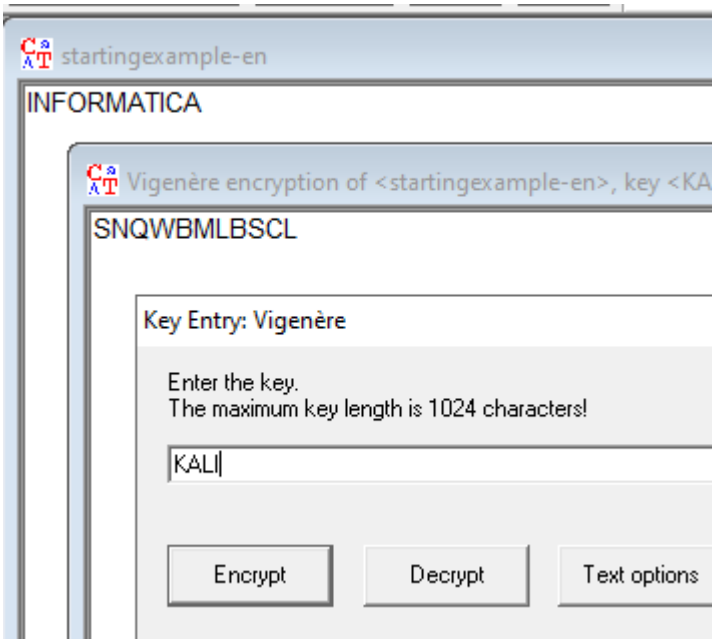
1. Escribe:
2. CRYPTOOL
3. Menú: **Encrypt / Decrypt → Symmetric (classic) → Substitution (Monoalphabetic)**.
4. Usa la clave automática que propone CrypTool.
5. Anota el texto cifrado.

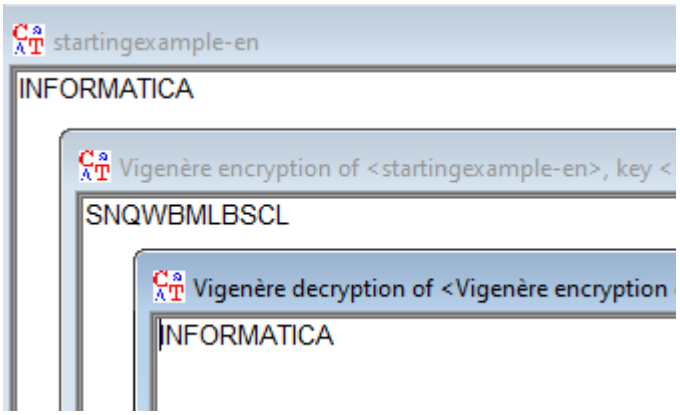


🔷 4. Vigenère sencillo

Objetivo: usar un cifrado polialfabético.

1. Escribe:
2. INFORMATICA
3. Menú: **Encrypt / Decrypt → Symmetric (classic) → Vigenère**.
4. Escribe como clave:
5. KALI
6. Cifra y luego descifra.



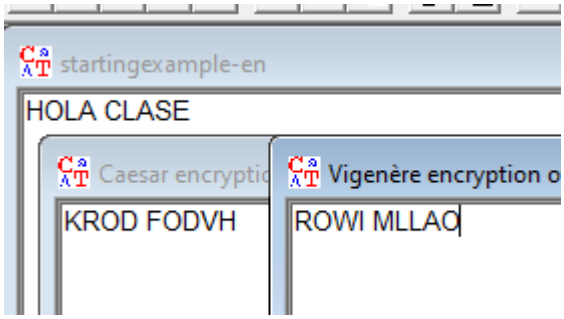


👉 **Resultado esperado:** comprobar que con la misma clave se puede recuperar el original.

🔷 5. Comparar César vs Vigenère

Objetivo: comparar dificultad.

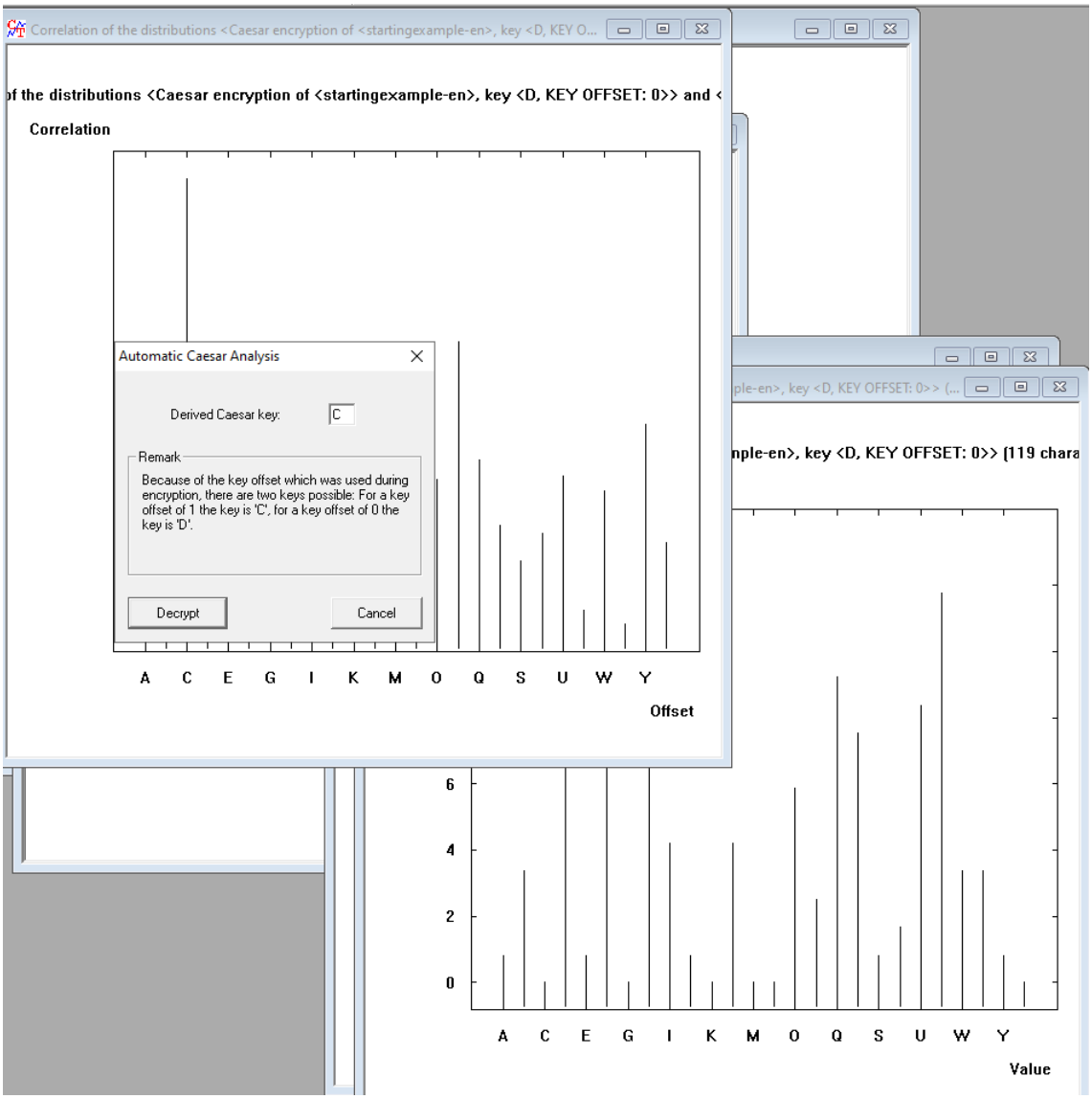
1. Escribe:
2. HOLA CLASE
3. Cifra con César (clave = 3).
4. Cifra con Vigenère (clave = KALI).
5. Compara visualmente los dos textos cifrados.



🔷 6. Análisis de frecuencias

Objetivo: detectar patrones.

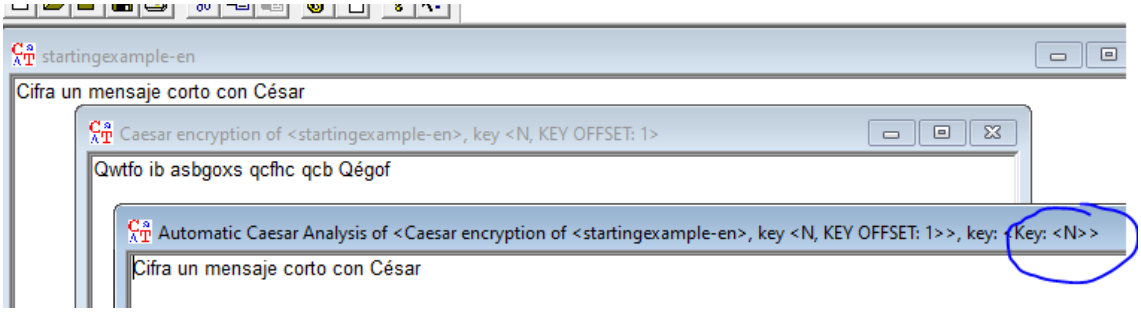
1. Cifra un texto largo (al menos 3 frases) con **César**.
2. Menú: **Analysis → Tools for Analysis → Frequency Analysis**.
3. Observa qué letras aparecen con más frecuencia.



🔷 7. Romper un César automáticamente

Objetivo: practicar criptoanálisis automático.

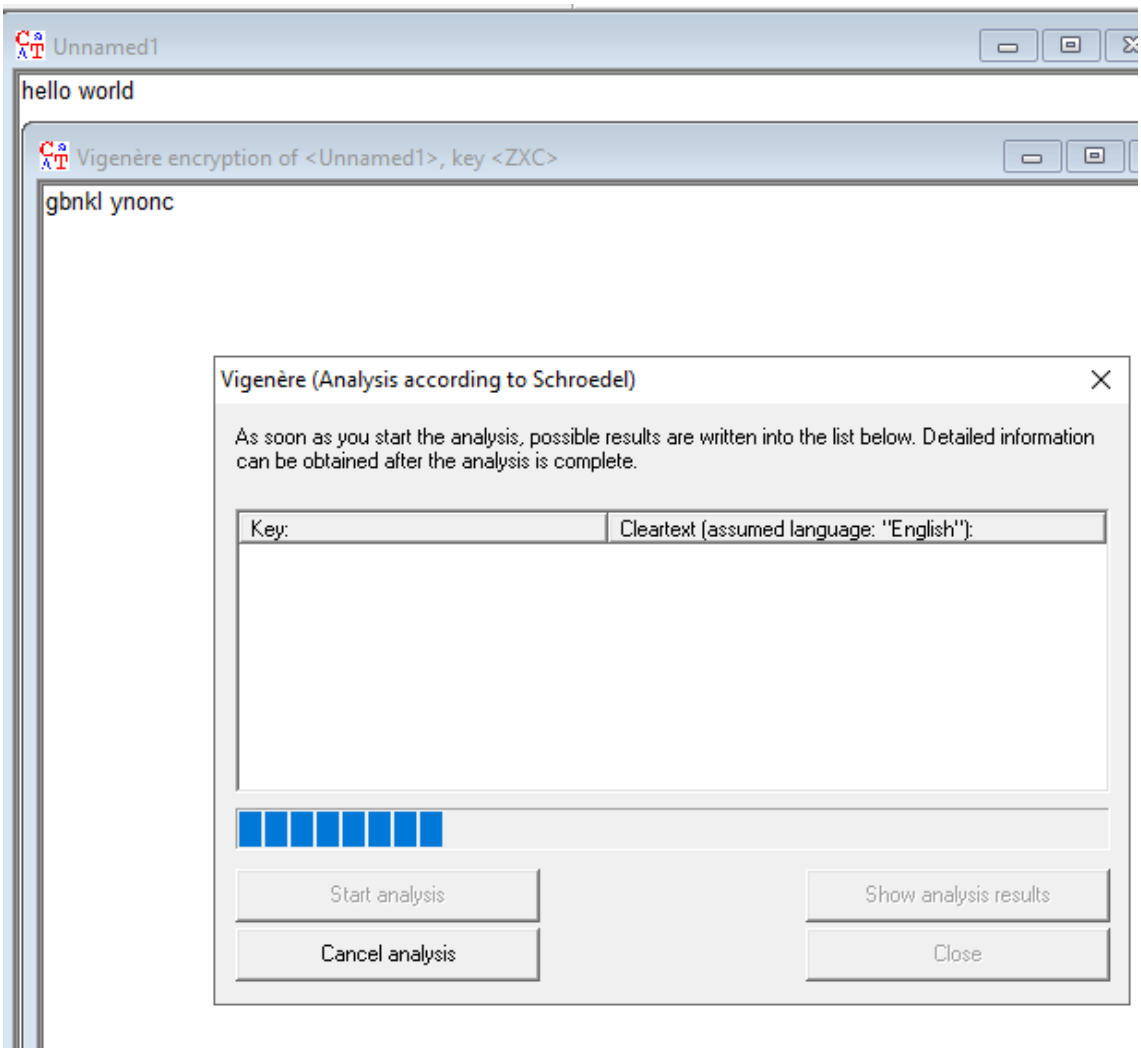
1. Cifra un mensaje corto con **César** (clave al azar).
2. Menú: **Analysis → Symmetric Encryption (classic) → Caesar**.
3. Usa el asistente de ataque: prueba todas las claves.
4. Encuentra cuál devuelve el mensaje original.

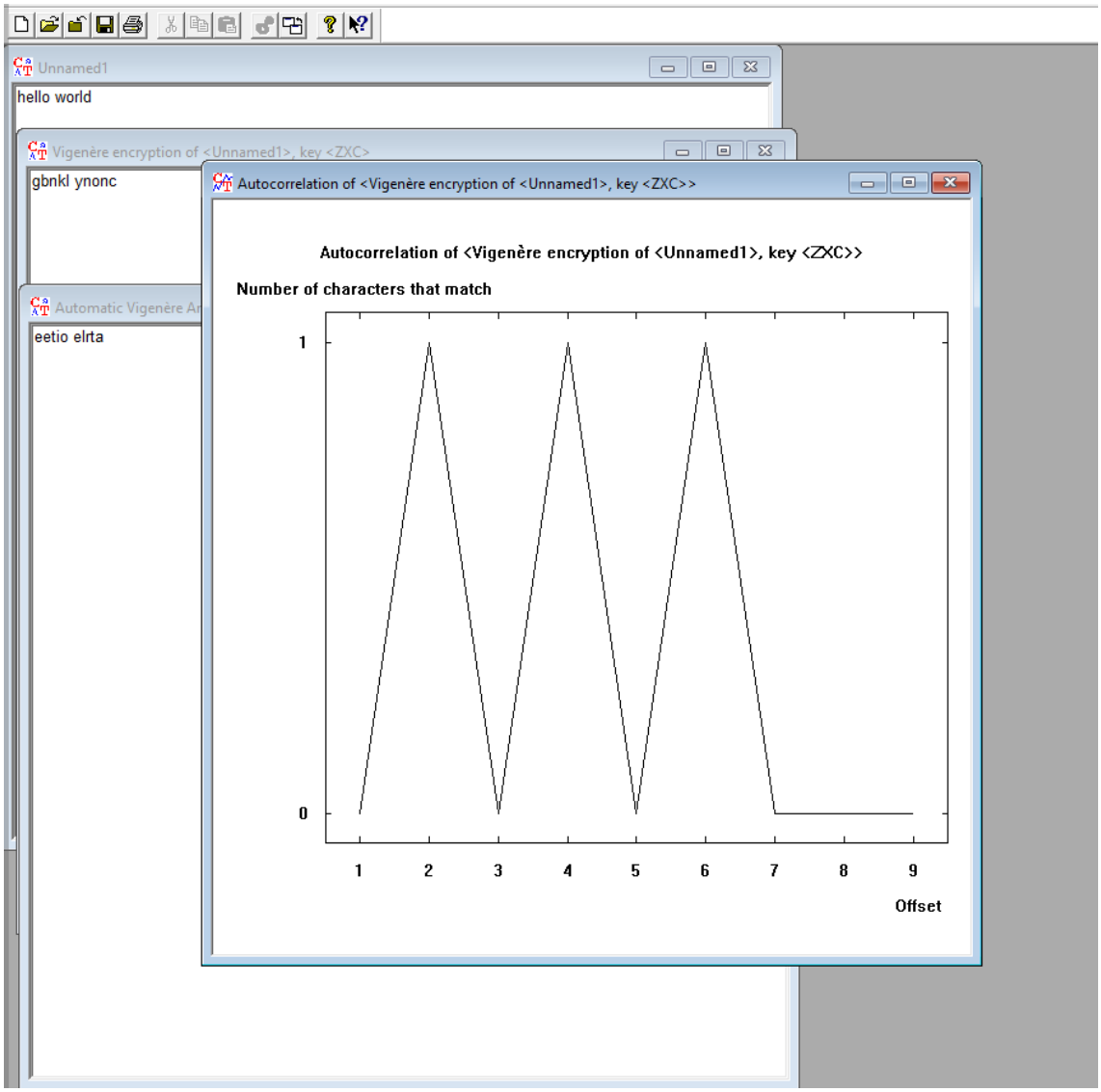


🔷 8. Romper un Vigenère

Objetivo: ver que también se puede atacar.

1. Cifra un texto corto con **Vigenère** (clave de 3 letras).
2. Menú: **Analysis → Symmetric Encryption (classic) → Vigenère**.
3. Usa la función de ataque con análisis de frecuencia.
4. Anota si logra encontrar la clave.



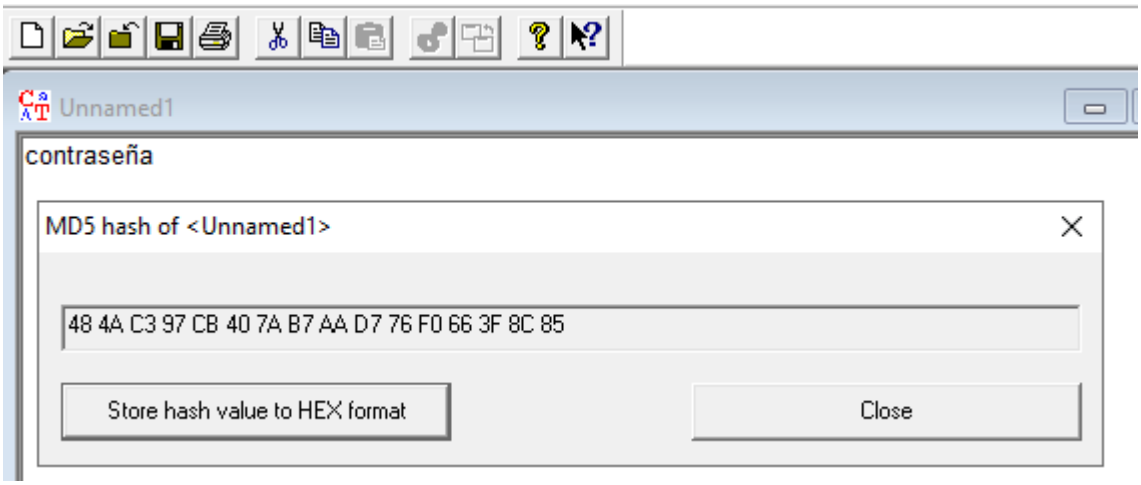


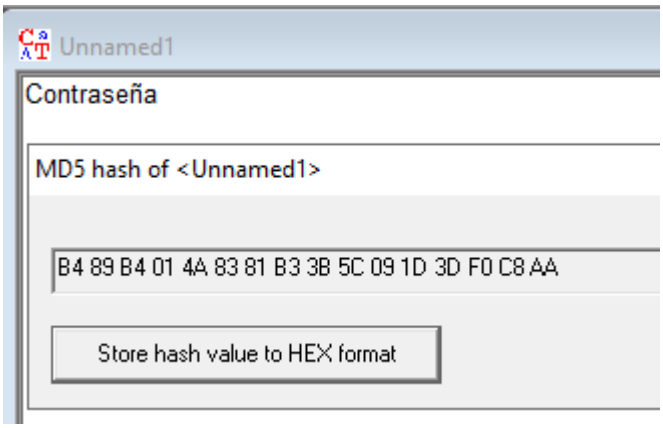
🔷 9. Hash con MD5

Objetivo: ver el resumen de un mensaje.

1. Escribe:
2. contraseña
3. Menú: **Analysis → Hash → Calculate MD5**.
4. Copia el hash obtenido.
5. Cambia el texto a:
6. Contraseña

y calcula de nuevo.



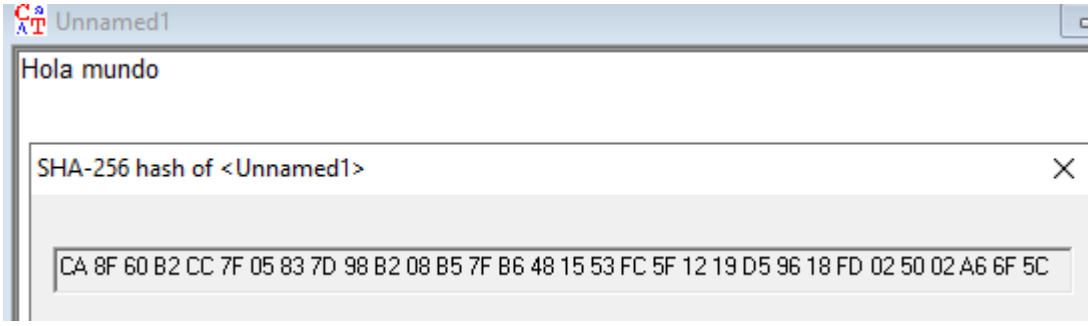


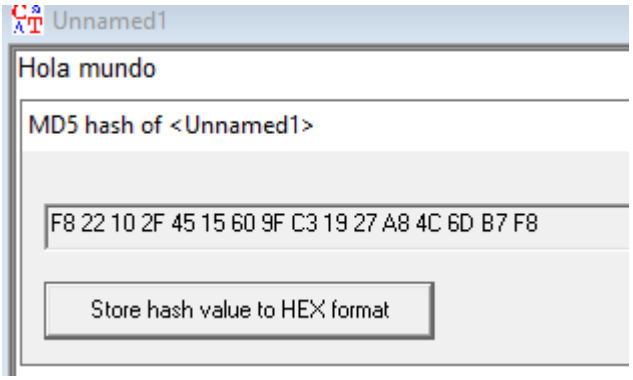
👉 **Resultado esperado:** observar que cambia por completo.

🔷 10. Hash con SHA-256

Objetivo: comparar con otro algoritmo.

1. Escribe:
2. Hola mundo
3. Menú: **Analysis → Hash → Calculate SHA (SHA-256)**.
4. Copia el hash.
5. Compara longitud y formato con el de MD5.



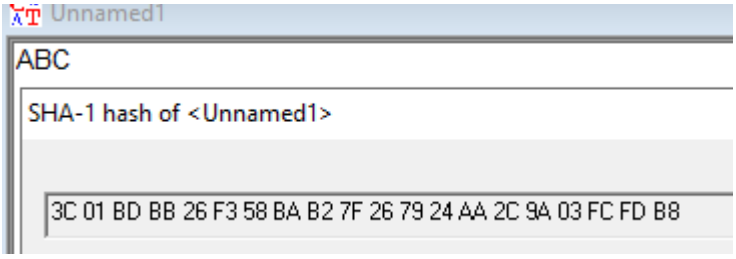


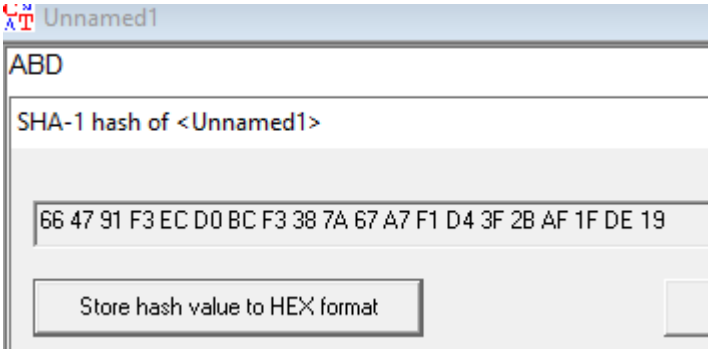
🔷 11. Efecto avalancha

Objetivo: comprobar cómo un pequeño cambio altera todo.

1. Calcula el hash SHA-1 de ABC.
2. Calcula el hash SHA-1 de ABD.
3. Compara los dos resultados.

👉 **Resultado esperado:** ver que todo el hash es distinto aunque solo cambió una letra.

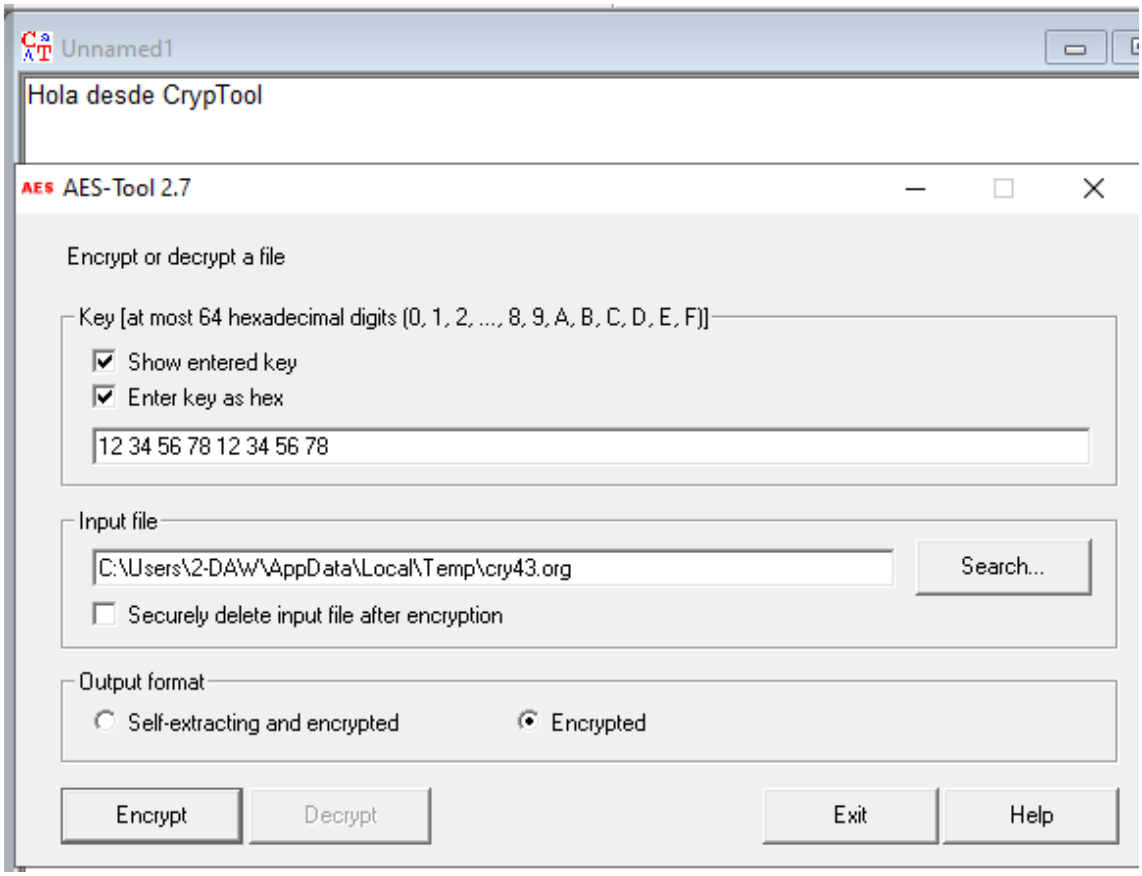


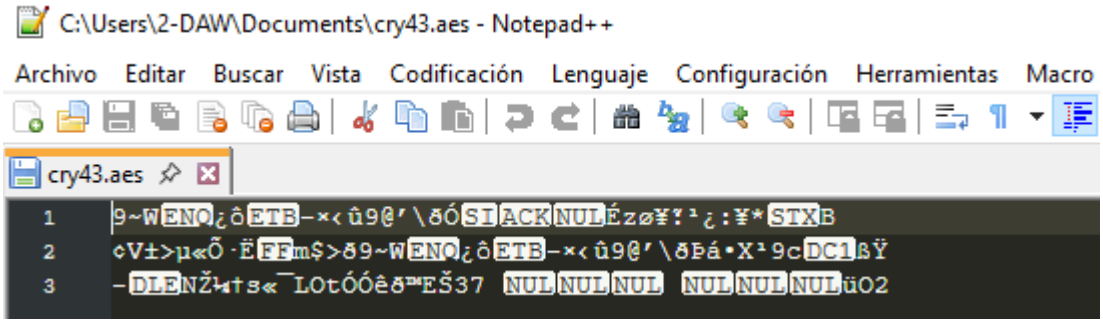


🔷 12. AES 128 bits

Objetivo: probar un cifrado moderno.

1. Escribe:
2. Hola desde CrypTool
3. Menú: **Encrypt / Decrypt → Symmetric (modern) → AES**.
4. Clave = 1234567812345678 (16 caracteres).
5. Anota el texto cifrado.

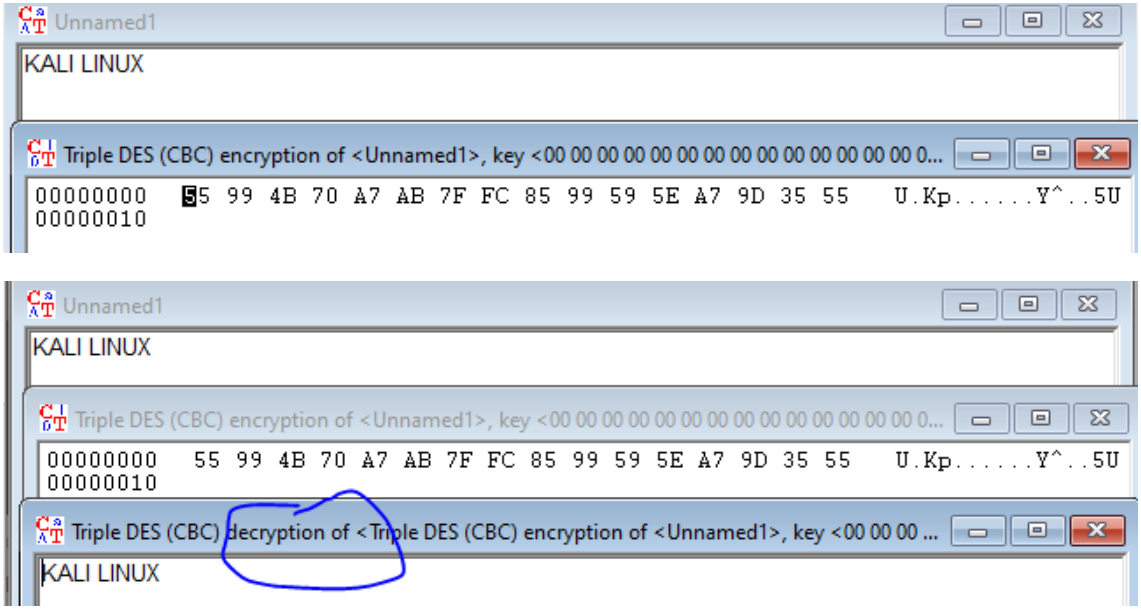




🔷 13. DES

Objetivo: comparar con AES.

1. Escribe:
2. KALI LINUX
3. Menú: **Encrypt / Decrypt → Symmetric (modern) → DES**.
4. Usa clave por defecto.
5. Compara el tamaño y formato del cifrado con el de AES.



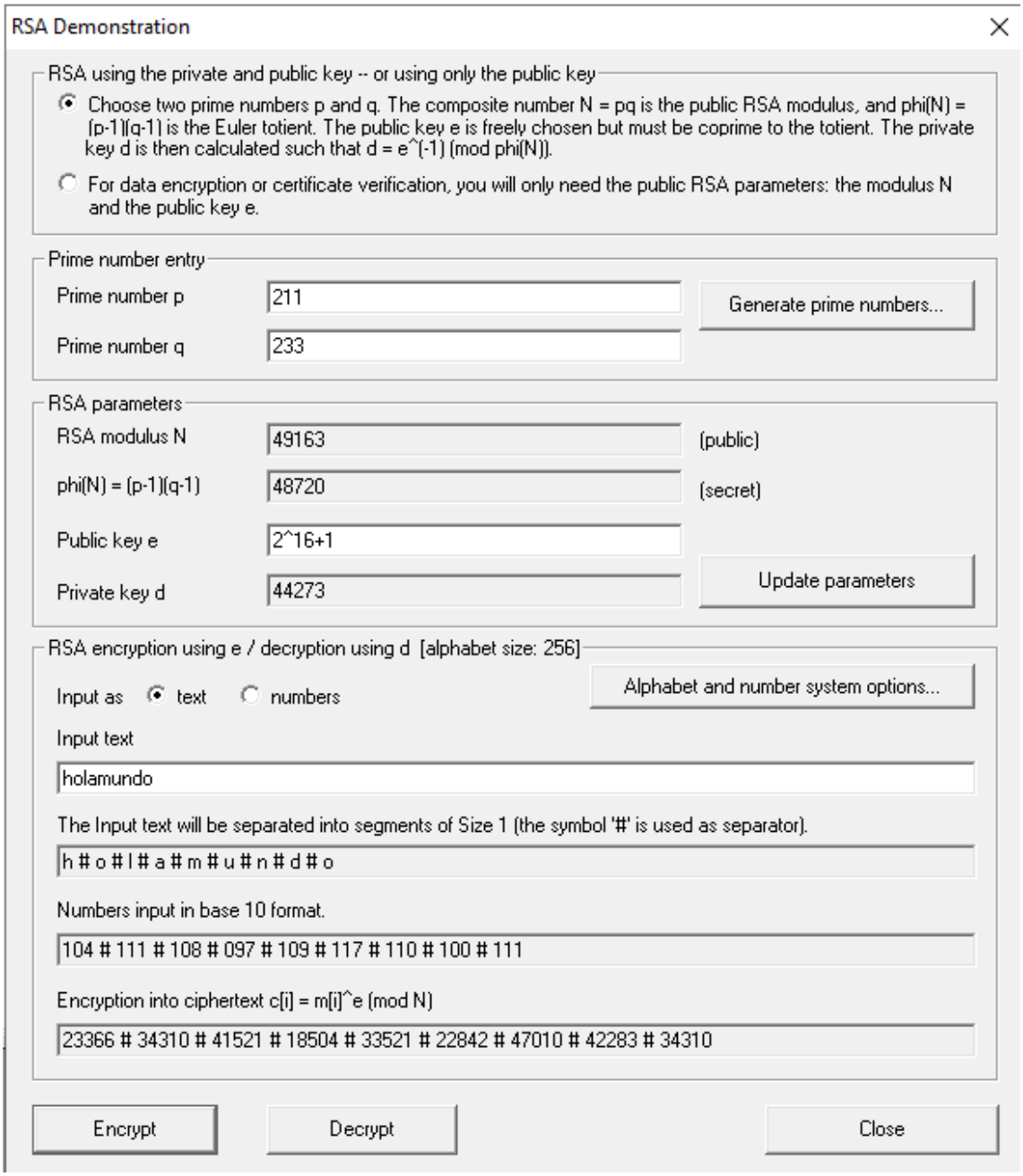
🔷 14. RSA (asimétrico)

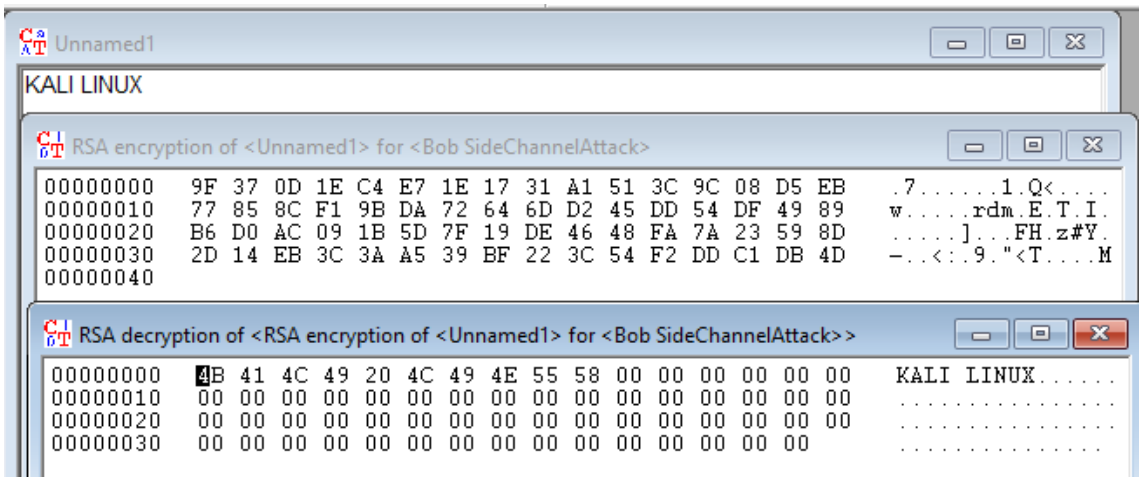
Objetivo: probar cifrado con claves pública/privada.

1. Menú: **Encrypt / Decrypt → Asymmetric → RSA Demonstration**.
2. Genera un par de claves.

3. Escribe un mensaje y **cifra con la clave pública**.

4. Descifra con la clave privada.



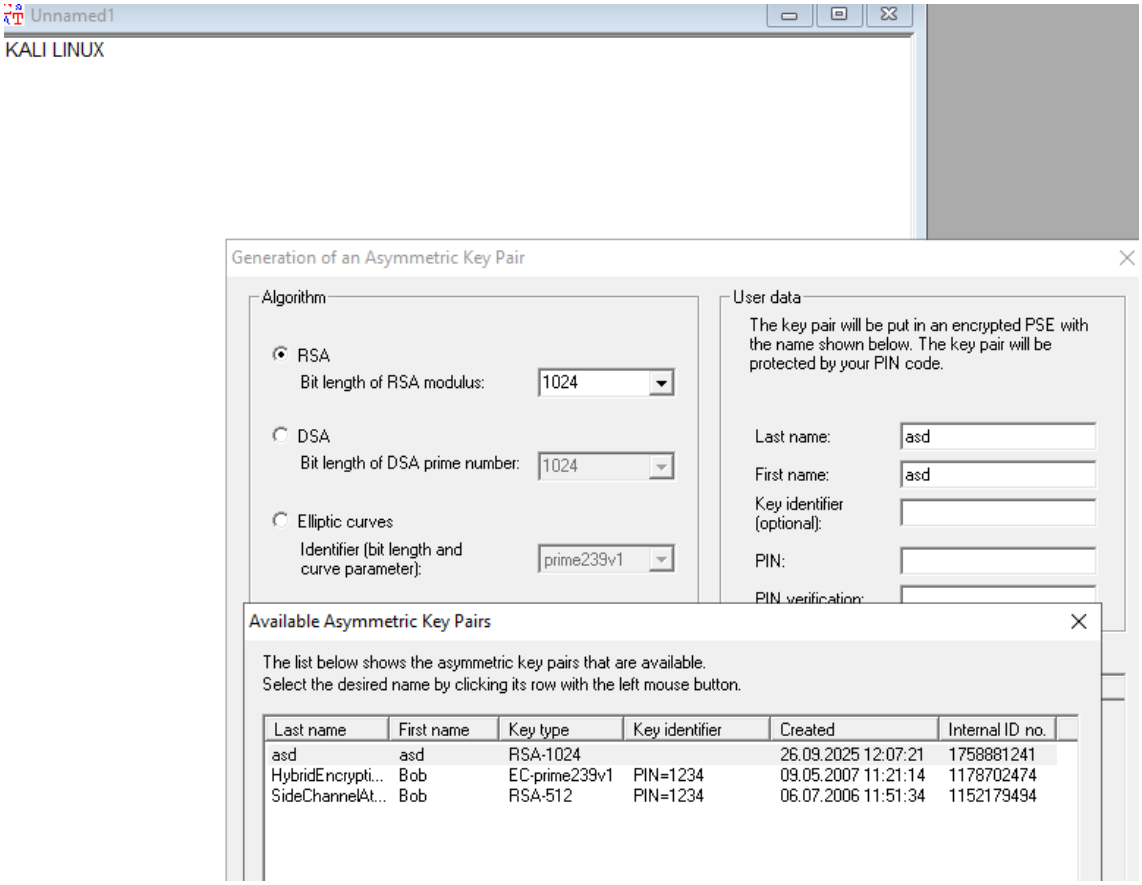


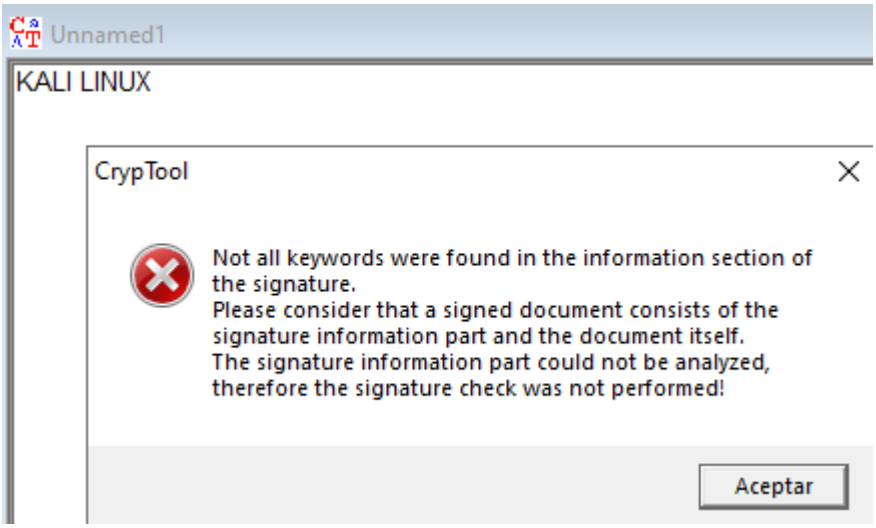
🔷 15. Firma digital simulada

Objetivo: entender autenticidad.

1. Menú: **Digital Signatures → Create/Verify Signature**.
2. Escribe un mensaje corto.
3. Firma con una clave privada.
4. Verifica con la clave pública.

👉 **Resultado esperado:** comprobar que solo con la clave pública se valida la firma.


Unnamed1
KALI LINUX
Generation of an Asymmetric Key Pair
Algorithm
RSA
Bit length of RSA modulus:
1024
DSA
Bit length of DSA prime number:
1024
Elliptic curves
Identifier (bit length and curve parameter):
prime239v1
User data
The key pair will be put in an encrypted PSE with the name shown below. The key pair will be protected by your PIN code.
Last name:
asd
First name:
asd
Key identifier (optional):
PIN:
PIN verification:
Available Asymmetric Key Pairs
The list below shows the asymmetric key pairs that are available.
Select the desired name by clicking its row with the left mouse button.
Last name
First name
Key type
Key identifier
Created
Internal ID no.
asd
asd
RSA-1024
26.09.2025 12:07:21
1758881241
HybridEncrypti...
Bob
EC-prime239v1
PIN=1234
09.05.2007 11:21:14
1178702474
SideChannelAt...
Bob
RSA-512
PIN=1234
06.07.2006 11:51:34
1152179494


Unnamed1
KALI LINUX
CrypTool
Not all keywords were found in the information section of the signature.
Please consider that a signed document consists of the signature information part and the document itself.
The signature information part could not be analyzed, therefore the signature check was not performed!
Aceptar