OPENSSL

METÓDO RAND

Herramienta de generación de números pseudoaleatorios con OpenSSL

NOMBRE

openssl rand → Genera datos pseudoaleatorios con OpenSSL.

DESCRIPCIÓN

El comando openssl rand genera datos pseudoaleatorios, útiles para claves criptográficas, tokens de sesión y otros datos de seguridad.

EJECUCIÓN Y SINTAXI

openssl rand [-out archivo] [-hex|-base64] [-engine motor] nobytes:

- **nobytes**: Número de bytes a generar.
- **-out archivo**: Especifica un archivo para guardar la salida.
- **-hex**: Formato de salida en hexadecimal.
- **-base64**: Formato de salida en Base64.
- **-engine motor:** Especifica el motor de hardware que se utilizará para la generación de números aleatorios, si está disponible.

PLANTEAMIENTO DE ACTIVIDAD

Generación y Gestión de Claves para Cifrado y Firma Digital en un Sistema de Mensajería Segura

Objetivo:

Implementar un sistema de mensajería segura utilizando claves criptográficas generadas con openssl rand. Aprenderemos a aplicar conceptos de criptografía simétrica y asimétrica, generación de claves seguras, y autenticación de mensajes.

Escenario:

Somos parte del equipo de seguridad de una empresa que necesita proteger su sistema de mensajería para evitar que los mensajes sean interceptados, alterados o enviados por usuarios no autorizados. Para ello, implementaremos un sistema de cifrado y firma digital basado en claves generadas con OpenSSL.

DESENVOLUPAMIENTO DE ACTIVIDAD

USUARIO BONILLA

Generación de Claves de Usuario y Configuración Inicial:

- Usamos *openssl* rand para crear una clave de 64 bytes en formato hexadecimal y guardamos esta clave en un archivo llamado *clave_simetrica.hex*.

  openssl rand -hex 64 -out clave_simetrica.hex

```
bonilla@bonilla ~/D/openssl_keys> openssl rand -hex 64 > clave_simetrica.hex

bonilla@bonilla ~/D/openssl_keys> ll
total 4,0K
-rw-rw-r-- 1 bonilla bonilla 129 nov 14 19:11 clave_simetrica.hex
bonilla@bonilla ~/D/openssl_keys>
```

Generación de un par de claves asimétricas (pública y privada) para cada usuario:

- Generamos el par de claves RSA de 2048 bits para Bonilla y guárdalos en los archivos *bonilla_privada.pem* y *bonilla_publica.pem*.

  openssl genpkey -algorithm RSA -out bonilla_privada.pem -pkeyopt rsa_keygen_bits:2048

```
bonilla@bonilla ~/D/openssl_keys> openssl genpkey -algorithm RSA -out bonilla_pr
ivada.pem -pkeyopt rsa_keygen_bits:2048
.......+.......+..+.++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++*..+...+.+.....+..........+.........++++++++++++++++++++++++++++++++++++++
++++++++++++++++++++++++++*..+..+..........+..............+......+..............
..............+..............+...+..+......+...+....+......+.....+....+..........
..+...+..............+...++++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++
...+..........+...+..+.........++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++++++++*.............+...+..+.+......+...+......+.....+......+...+......
.+....+..+.+.....+....++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
+++++++*.......+...+......+.....+....+...+..+.+..+....+............+..............
.........+..+...+.......+........+.........+.....+...+...+.....................+.....
.+....+..+.........+...+.......++++++++++++++++++++++++++++++++++++++++++++++++++
++++++++++++++++
bonilla@bonilla ~/D/openssl_keys> ll
total 8,0K
-rw------- 1 bonilla bonilla 1,7K nov 14 19:12 bonilla_privada.pem
-rw-rw-r-- 1 bonilla bonilla  129 nov 14 19:11 clave_simetrica.hex
bonilla@bonilla ~/D/openssl_keys>
```

openssl rsa -pubout -in bonilla_privada.pem -out bonilla_publica.pem

```
bonilla@bonilla ~/D/openssl_keys> openssl rsa -pubout -in bonilla_privada.pem -o
ut bonilla_publica.pem
writing RSA key
bonilla@bonilla ~/D/openssl_keys> ll
total 12K
-rw------- 1 bonilla bonilla 1,7K nov 14 19:12 bonilla_privada.pem
-rw-rw-r-- 1 bonilla bonilla  451 nov 14 19:12 bonilla_publica.pem
-rw-rw-r-- 1 bonilla bonilla  129 nov 14 19:11 clave_simetrica.hex
```

- Repetimos los pasos anteriores para Rubén, y guardamos sus claves en *ruben_privada.pem* y *ruben_publica.pem*.

  openssl genpkey -algorithm RSA -out ruben_privada.pem -pkeyopt rsa_keygen_bits:2048

```
cliente@carlosbonilla ~/openssl_keys> openssl genpkey -algorithm RSA -out ruben_
privada.pem -pkeyopt rsa_keygen_bits:2048
...+.....+.+.........+..........+...+...+..+..........+.....+...+.....+.....
.+.....++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*.+...+
....+.....+.....+....+.....+.+...+..............+.+..+........+........+........
..+.....+....++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*.
.+.+........+...+...+....+............+...+........+...+.....+....+...........
..+...+...+.........+.+..+.......+..+.+.........+..........+.........+.+..+...+
....++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
..+........+...+............+...+.....+...+.+..+.........+..........+...+.....
+..+.........++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*.
+..+............+....+..+...+.....+....+..+.......+..+...+.+.....+...+.....+++
++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++*.....+.+.......
.+.+.....+.....+.....+....+...+..+....+.....+.....+.....+..............+...
....+..+.....................+..+....+.....+..+.+..........+.........+...+..+..
.......+.....+.+.........+.......+..+.+.....+........+.+..+....+.............+
.........+............+..+.......+...+...+.........+...+...+.....+.......+..+..
....+....+...+.....+...........+..+.....+.....+...+...........+...........
......++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
cliente@carlosbonilla ~/openssl_keys> ls
ruben_privada.pem
cliente@carlosbonilla ~/openssl_keys>
```

openssl rsa -pubout -in ruben_privada.pem -out ruben_publica.pem

```
cliente@carlosbonilla ~/openssl_keys> openssl rsa -pubout -in ruben_privada.pem
-out ruben_publica.pem
writing RSA key
cliente@carlosbonilla ~/openssl_keys> ll
total 8,0K
-rw------- 1 cliente cliente 1,7K nov 14 19:13 ruben_privada.pem
-rw-rw-r-- 1 cliente cliente  451 nov 14 19:14 ruben_publica.pem
cliente@carlosbonilla ~/openssl_keys>
```

Envío de Mensajes Cifrados y Firmados:

- Creamos un mensaje en un archivo llamado *mensaje.txt* con el texto “Mensaje cifrado para Rubensito con Samba”.

```
bonilla@bonilla ~/D/openssl_keys> touch mensaje.txt
bonilla@bonilla ~/D/openssl_keys> echo 'Mensaje cifrado para Rubensito con Samba
' > mensaje.txt
bonilla@bonilla ~/D/openssl_keys> ls
bonilla_privada.pem  bonilla_publica.pem  clave_simetrica.hex  mensaje.txt
bonilla@bonilla ~/D/openssl_keys> cat mensaje.txt
Mensaje cifrado para Rubensito con Samba
bonilla@bonilla ~/D/openssl_keys>
```

- Utilizamos la clave simétrica generada anteriormente para cifrar el mensaje usando AES-256-CBC y guarda el mensaje cifrado en *mensaje_cifrado.enc*.

  openssl enc -aes-256-cbc -in mensaje.txt -out mensaje_cifrado.enc -pass file:clave_simetrica.hex

```
bonilla@bonilla ~/D/openssl_keys> openssl enc -aes-256-cbc -in mensaje.txt -out
mensaje_cifrado.enc -pass file:clave_simetrica.hex
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
bonilla@bonilla ~/D/openssl_keys> ll
total 20K
-rw------- 1 bonilla bonilla 1,7K nov 14 19:12 bonilla_privada.pem
-rw-rw-r-- 1 bonilla bonilla  451 nov 14 19:12 bonilla_publica.pem
-rw-rw-r-- 1 bonilla bonilla  129 nov 14 19:11 clave_simetrica.hex
-rw-rw-r-- 1 bonilla bonilla   64 nov 14 19:19 mensaje_cifrado.enc
-rw-rw-r-- 1 bonilla bonilla   41 nov 14 19:18 mensaje.txt
bonilla@bonilla ~/D/openssl_keys>
```

Firmar el mensaje cifrado:

- Usamos la clave privada de Bonilla (*bonilla_privada.pem*) para crear una firma digital del mensaje cifrado y la almacenamos en un archivo llamado *firma_mensaje.bin.*

  openssl dgst -sha256 -sign bonilla_privada.pem -out firma_mensaje.bin mensaje_cifrado.enc

```
bonilla@bonilla ~/D/openssl_keys> openssl dgst -sha256 -sign bonilla_privada.pem -o
ut firma_mensaje.bin mensaje_cifrado.enc
bonilla@bonilla ~/D/openssl_keys> # Creamos la firma digital con mi clave privada
```

Compartir la clave simétrica de forma segura:

- Recibimos la clave pública de Ruben (*ruben_publica.pem*).

```
bonilla@bonilla ~/D/openssl_keys> sudo scp -v root@192.168.1.138:/home/cliente/open
ssl_keys/ruben_publica.pem .
Executing: program /usr/bin/ssh host 192.168.1.138, user root, command sftp
OpenSSH_9.6p1 Ubuntu-3ubuntu13.5, OpenSSL 3.0.13 30 Jan 2024
```

```
bonilla@bonilla ~/D/openssl_keys> ls
bonilla_privada.pem  clave_simetrica.hex  mensaje_cifrado.enc  ruben_publica.pem
bonilla_publica.pem  firma_mensaje.bin    mensaje.txt
```

- Ciframos el archivo *clave_simetrica.hex* con la clave pública de Rubén para que solo él pueda descifrar. Guardamos el resultado en *clave_simetrica_cifrada.enc*.

  openssl rsautl -encrypt -inkey ruben_publica.pem -pubin -in clave_simetrica.hex -out clave_simetrica_cifrada.enc

```
bonilla@bonilla ~/D/openssl_keys> openssl rsautl -encrypt -inkey ruben_publica.pem
-pubin -in clave_simetrica.hex -out clave_simetrica_cifrada.enc
The command rsautl was deprecated in version 3.0. Use 'pkeyutl' instead.
bonilla@bonilla ~/D/openssl_keys> ls
bonilla_privada.pem          clave_simetrica.hex  mensaje.txt
bonilla_publica.pem          firma_mensaje.bin    ruben_publica.pem
clave_simetrica_cifrada.enc  mensaje_cifrado.enc
bonilla@bonilla ~/D/openssl_keys>
```

USUARIO RUBEN

Estos son los archivos que tenemos ahora mismo en el user Bonilla:

```
bonilla@bonilla ~/D/openssl_keys> ll
total 32K
-rw------- 1 bonilla bonilla 1,7K nov 14 19:12 bonilla_privada.pem
-rw-rw-r-- 1 bonilla bonilla  451 nov 14 19:12 bonilla_publica.pem
-rw-rw-r-- 1 bonilla bonilla  256 nov 14 19:25 clave_simetrica_cifrada.enc
-rw-rw-r-- 1 bonilla bonilla  129 nov 14 19:11 clave_simetrica.hex
-rw-rw-r-- 1 bonilla bonilla  256 nov 14 19:20 firma_mensaje.bin
-rw-rw-r-- 1 bonilla bonilla   64 nov 14 19:19 mensaje_cifrado.enc
-rw-rw-r-- 1 bonilla bonilla   41 nov 14 19:18 mensaje.txt
-rw-r--r-- 1 root    root     451 nov 14 19:23 ruben_publica.pem
bonilla@bonilla ~/D/openssl_keys>
```

Descifrar la clave simétrica:

- Recibimos la clave simetrica que hemos cifrado con la clave pública de Ruben.

```
cliente@carlosbonilla ~/openssl_keys> sudo scp -v root@192.168.1.43:/home/bonilla/Document
os/openssl_keys/clave_simetrica_cifrada.enc .
Executing: program /usr/bin/ssh host 192.168.1.43, user root, command scp -v -f /home/boni
lla/Documentos/openssl_keys/clave_simetrica_cifrada.enc
OpenSSH_8.9p1 Ubuntu-3ubuntu0.10, OpenSSL 3.0.2 15 Mar 2022
debug1: Reading configuration data /etc/ssh/ssh_config
debug1: /etc/ssh/ssh_config line 19: include /etc/ssh/ssh_config.d/*.conf matched no files
debug1: /etc/ssh/ssh_config line 21: Applying options for *
debug1: Connecting to 192.168.1.43 [192.168.1.43] port 22.
debug1: Connection established.
debug1: identity file /root/.ssh/id_rsa type -1
```

```
cliente@carlosbonilla ~/openssl_keys> ll
total 12K
-rw-r--r-- 1 root    root     256 nov 14 19:29 clave_simetrica_cifrada.enc
-rw------- 1 cliente cliente 1,7K nov 14 19:13 ruben_privada.pem
-rw-rw-r-- 1 cliente cliente  451 nov 14 19:14 ruben_publica.pem
cliente@carlosbonilla ~/openssl_keys>
```

- Usamos su clave privada (*ruben_privada.pem*) para descifrar la clave simétrica y guardarla en clave_simetrica_ruben.hex

  openssl rsautl -decrypt -inkey ruben_privada.pem -in clave_simetrica_cifrada.enc -out clave_simetrica_ruben.hex

```
cliente@carlosbonilla ~/openssl_keys> openssl rsautl -decrypt -inkey ruben_privada.pem -in
 clave_simetrica_cifrada.enc -out clave_simetrica_ruben.hex
The command rsautl was deprecated in version 3.0. Use 'pkeyutl' instead.
cliente@carlosbonilla ~/openssl_keys> ll
total 16K
-rw-r--r-- 1 root    root     256 nov 14 19:29 clave_simetrica_cifrada.enc
-rw-rw-r-- 1 cliente cliente  129 nov 14 19:31 clave_simetrica_ruben.hex
-rw------- 1 cliente cliente 1,7K nov 14 19:13 ruben_privada.pem
-rw-rw-r-- 1 cliente cliente  451 nov 14 19:14 ruben_publica.pem
cliente@carlosbonilla ~/openssl_keys>
```

Descifrar el mensaje:

- Recibimos el mensaje cifrado del user Bonilla.

```
cliente@carlosbonilla ~/openssl_keys> sudo scp -v root@192.168.1.43:/home/bonilla/Document
os/openssl_keys/mensaje_cifrado.enc .
Executing: program /usr/bin/ssh host 192.168.1.43, user root, command scp -v -f /home/boni
lla/Documentos/openssl_keys/mensaje_cifrado.enc
OpenSSH_8.9p1 Ubuntu-3ubuntu0.10, OpenSSL 3.0.2 15 Mar 2022
debug1: Reading configuration data /etc/ssh/ssh_config
debug1: /etc/ssh/ssh_config line 19: include /etc/ssh/ssh_config.d/*.conf matched no files
debug1: /etc/ssh/ssh_config line 21: Applying options for *
debug1: Connecting to 192.168.1.43 [192.168.1.43] port 22.
debug1: Connection established.
```

- Con la clave simétrica descifrada, podemos descifrar el mensaje original usando AES-256-CBC, y guardamos el mensaje en *mensaje_descifrado.txt* en el user Ruben.

  openssl enc -d -aes-256-cbc -in mensaje_cifrado.enc -out mensaje_descifrado.txt -pass file:clave_simetrica_ruben.hex

```
cliente@carlosbonilla ~/openssl_keys> openssl enc -d -aes-256-cbc -in mensaje_cifrado.enc
-out mensaje_descifrado.txt -pass file:clave_simetrica_ruben.hex
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
cliente@carlosbonilla ~/openssl_keys> ll
total 24K
-rw-r--r-- 1 root    root     256 nov 14 19:29 clave_simetrica_cifrada.enc
-rw-rw-r-- 1 cliente cliente  129 nov 14 19:31 clave_simetrica_ruben.hex
-rw-r--r-- 1 root    root      64 nov 14 19:33 mensaje_cifrado.enc
-rw-rw-r-- 1 cliente cliente   41 nov 14 19:34 mensaje_descifrado.txt
-rw------- 1 cliente cliente 1,7K nov 14 19:13 ruben_privada.pem
-rw-rw-r-- 1 cliente cliente  451 nov 14 19:14 ruben_publica.pem
cliente@carlosbonilla ~/openssl_keys>
```

- Vemos que el contenido del mensaje es correcto.

```
cliente@carlosbonilla ~/openssl_keys> cat mensaje_descifrado.txt
Mensaje cifrado para Rubensito con Samba
cliente@carlosbonilla ~/openssl_keys> #TENEMOS EL MENSAJE DESCIFRADO
```

Verificar la firma del mensaje:

- Recibimos la clave pública de Bonilla.

```
cliente@carlosbonilla ~/openssl_keys> sudo scp -v root@192.168.1.43:/home/bonilla/Document
os/openssl_keys/bonilla_publica.pem .
Executing: program /usr/bin/ssh host 192.168.1.43, user root, command scp -v -f /home/boni
```

- Recibimos la firma.

```
cliente@carlosbonilla ~/openssl_keys> sudo scp -v root@192.168.1.43:/home/bonilla/Document
os/openssl_keys/firma_mensaje.bin .
Executing: program /usr/bin/ssh host 192.168.1.43, user root, command scp -v -f /home/boni
lla/Documentos/openssl_keys/firma_mensaje.bin
OpenSSH_8.9p1 Ubuntu-3ubuntu0.10, OpenSSL 3.0.2 15 Mar 2022
```

- Verificamos la autenticidad del mensaje cifrado utilizando la clave pública de Bonilla (*bonilla_publica.pem*)

  openssl dgst -sha256 -verify bonilla_publica.pem -signature firma_mensaje.bin mensaje_cifrado.enc

```
cliente@carlosbonilla ~/openssl_keys> ll
total 32K
-rw-r--r-- 1 root    root     451 nov 14 19:37 bonilla_publica.pem
-rw-r--r-- 1 root    root     256 nov 14 19:29 clave_simetrica_cifrada.enc
-rw-rw-r-- 1 cliente cliente  129 nov 14 19:31 clave_simetrica_ruben.hex
-rw-r--r-- 1 root    root     256 nov 14 19:39 firma_mensaje.bin
-rw-r--r-- 1 root    root      64 nov 14 19:33 mensaje_cifrado.enc
-rw-rw-r-- 1 cliente cliente   41 nov 14 19:34 mensaje_descifrado.txt
-rw------- 1 cliente cliente 1,7K nov 14 19:13 ruben_privada.pem
-rw-rw-r-- 1 cliente cliente  451 nov 14 19:14 ruben_publica.pem
cliente@carlosbonilla ~/openssl_keys> openssl dgst -sha256 -verify bonilla_publica.pem -si
gnature firma_mensaje.bin mensaje_cifrado.enc
Verified OK
cliente@carlosbonilla ~/openssl_keys>
```