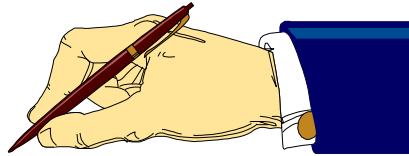


Firme digitali con OpenSSL

Alfredo De Santis



Dipartimento di Informatica
Università di Salerno

ads@unisa.it

<http://www.dia.unisa.it/professori/ads>



Aprile 2017

Sommario

- Firma RSA
 - Firma `rsautl -sign`
 - Verifica `rsautl -verify`
- Firma DSA
 - Generazione parametri `dsaparam`
 - Generazione chiavi `gensa`
 - Firma `dgst -sign`
 - Verifica `dgst -verify`
- Firma ECDSA
 - Generazione parametri `ecparam`
 - Generazione chiavi `ecparam -genkey`
 - Firma `dgst -sign`
 - Verifica `dgst -verify`

Firma RSA in OpenSSL

(Il Comando `rsautl`)

- Il comando `rsautl` consente di usare le chiavi RSA per la firma
 - Permette di specificare opzioni per firmare e verificare le firme
 - **N.B.** La firma è di solito apposta su hash di file
 - La dimensione dei dati da firmare è vincolata alla lunghezza della chiave

Firma RSA in OpenSSL

Opzioni principali del comando rsautl

```
openssl rsautl [options]
```

➤ options

- **-in file** File di input
- **-out file** File di output
- **-encrypt** Cifra con la chiave pubblica
- **-decrypt** Decifra con la chiave privata
- **-sign** Firma con la chiave privata
- **-verify** Verifica con la chiave pubblica
- **-inkey** Chiave presa in input
- **-passin arg** Sorgente da cui deve essere letta la password
- **-pubin** Specifica che l'input è una chiave pubblica RSA
- **-pkcs, -raw** Padding da usare: PKCS#1 v1.5 (default) o nessun padding

Firma RSA in OpenSSL

Opzioni principali del comando rsautl

```
openssl rsautl [options]
```

➤ options

- **-in file** File di input
- **-out file** File di output
- **-encrypt** Cifra con la chiave pubblica
- **-decrypt** Decifra con la chiave privata
- **-sign** Firma con la chiave privata
- **-verify** Verifica con la chiave pubblica
- **-inkey** Chiave presa in input
- **-passin arg** Sorgente da cui deve essere letta la password
- **-pubin** Specifica che l'input è una chiave pubblica RSA
- **-pkcs, -raw** Padding da usare: PKCS#1 v1.5 (default) o nessun padding

Firma RSA in OpenSSL

(Esempio Firma/Verifica)

Mediante il seguente comando è possibile firmare un file con RSA

- Usando la chiave privata contenuta in **rsaprivatekey.pem**, viene firmato il contenuto del file **testoInChiaro.txt** e la firma è scritta nel file **rsasign.bin**

```
openssl rsautl -sign -inkey rsaprivatekey.pem  
-in testoInChiaro.txt -out rsasign.bin
```

Mediante il seguente comando è possibile verificare la firma RSA di un file

- Usando la chiave pubblica contenuta in **rsapublickey.pem**, viene verificata la firma del file **testoInChiaro.txt**, contenuta in **rsasign.bin**
 - Se la verifica ha successo, il file **rsasign.bin** viene correttamente decifrato, altrimenti viene restituito un messaggio di errore

```
openssl rsautl -verify -pubin -inkey rsapublickey.pem  
-in rsasign.bin -out testoInChiaro.txt
```

Firma RSA in OpenSSL

(Esempio Firma/Verifica con Hash)

Mediante RSA è anche possibile firmare l'hash di un file

- Usando la chiave privata contenuta in **rsaprivatekey.pem**, viene firmato l'hash SHA1 del file **testoInChiaro.txt** e la firma è scritta nel file **rsasign.bin**

```
openssl sha1 -sign rsaprivatekey.pem -out rsasign.bin  
testoInChiaro.txt
```

Mediante il seguente comando è possibile verificare la firma RSA apposta sull'hash di un file

- Usando la chiave pubblica contenuta in **rsapublickey.pem** e l'algoritmo SHA1, è verificata la firma di **testoInChiaro.txt**, contenuta in **rsasign.bin**

```
openssl sha1 -verify rsapublickey.pem -signature rsasign.bin  
testoInChiaro.txt
```

Firma DSA in OpenSSL

(Generazione dei Parametri)

- È possibile generare i parametri dello schema DSA mediante il comando **dsaparam**

Opzioni principali del comando **dsaparam**

```
openssl dsaparam [options] [numbits]
```

- **options**

- **-inform arg** Formato di input, dove **arg** può essere DER o PEM
- **-outform arg** Formato di output, dove **arg** può essere DER o PEM
- **-in arg** Dove **arg** è il file di input
- **-out arg** Dove **arg** è il file di output
- **-text** Stampa i parametri DSA in formato testuale

- **numbits**

- Numero di bit da generare, di default sono 512

Firma DSA in OpenSSL

(Generazione dei Parametri)

- È possibile generare i parametri dello schema DSA mediante il comando **dsaparam**

Opzioni principali del comando **dsaparam**

```
openssl dsaparam [options] [numbits]
```

- **options**

- **-inform arg** Formato di input. arg può essere DER o PEM
- **-outform arg** Formato di output. Per ottenere la lista completa delle opzioni del comando **dsaparam** è possibile utilizzare **man dsaparam**
- **-in arg** Dove **arg** è il file di input
- **-out arg** Dove **arg** è il file di output
- **-text** Stampa i parametri DSA in formato testuale

- **numbits**

- Numero di bit da generare, di default sono 512

Firma DSA in OpenSSL

(Contenuto del file dsaparams.pem)

DSA-Parameters: (1024 bit)

p:

00:a0:d5:c6:c6:85:21:8a:fc:a5:90:b8:19:24:4b:
07:11:b6:6c:41:1f:3d:15:71:52:9c:d1:6e:9e:06:
27:1c:e5:fa:d5:90:ba:55:43:de:57:a6:71:58:1f:
08:20:61:9b:31:9e:c9:e8:c9:ba:d5:f6:02:ec:17:
ad:00:fb:55:c3:62:b6:c9:81:8a:68:74:ab:1b:17:
fa:83:37:61:b7:4d:6c:f3:f2:70:20:95:ec:f6:c1:
c9:b0:f0:f4:28:33:62:b4:56:64:9e:66:e0:72:77:
6f:e7:f1:ad:db:b4:8e:94:57:fa:4f:81:6e:69:b1:
d6:7c:ea:e6:84:53:9b:ea:dd

p

q:

00:b7:f3:a3:4a:6b:59:b0:06:71:29:f8:d2:30:0f:
cb:c9:63:75:2f:5b

q

g:

32:5b:6c:78:33:10:fa:0d:44:2b:4b:b9:56:08:cb:
fc:84:1b:11:47:10:63:a8:33:3e:0c:09:12:c8:66:
0b:71:9c:f9:65:8c:4c:36:f3:2d:94:7a:f2:c0:eb:
63:2d:44:cb:08:3e:91:ee:e8:51:12:7b:5a:98:37:
cf:a5:46:0f:b0:63:a6:c8:fe:4c:db:34:d5:61:f3:
31:ee:22:df:72:c1:dc:6b:21:9c:14:e0:9d:cb:1d:
3c:ff:77:6b:d4:fd:54:a1:df:fb:a8:41:23:ae:f2:
5c:c2:b0:43:32:0d:c7:f2:7d:69:3f:6f:e7:72:b8:
f0:1b:88:86:04:9d:53:c4

g

-----BEGIN DSA PARAMETERS-----

```
MIIBHgKBgQCg1cbGhSGK/KWQuBkkSwcRtmxBHz0VcVKc0W6eBicc5frVklpVQ95X
pnFYHwggYZsxnsnoybrV9gLSF60A+1XDYrbJgYpodKsbF/qDN2G3TWzz8nAglez2
wcmw8PQoM2K0VmSeZuByd2/n8a3btI6UV/pPgW5psdZ86uaEU5vq3QIVALfzo0pr
WbAGcSn40jAPy8ljds9bAoGAMltseDMQ+g1EK0u5VgjL/IQbEUcQY6gzPgwJEshm
C3Gc+WMMTDbzLZR68sDrYy1Eyw+ke7oURJ7Wpg3z6VGD7Bjpsj+TNs01WHzMe4i
33LB3GshnBTgncsdPP93a9T9VKhf+6hBI67yXMKwQzINx/J9aT9v53K48BuIhgSd
U8Q=
```

-----END DSA PARAMETERS-----

I parametri DSA in OpenSSL sono rappresentati secondo lo standard PKCS #8 (RFC5208)

➤ <https://tools.ietf.org/html/rfc5208>

Codifica PEM dei parametri DSA

Firma DSA in OpenSSL

(Esempio Generazione Chiavi)

Mediante il seguente comando è possibile generare una coppia di chiavi DSA, utilizzando i parametri contenuti nel file `dsaparams.pem`

- Tale coppia sarà cifrata mediante 3DES
- Il risultato verrà scritto nel file `dsaprivatekey.pem`

```
openssl gendsa -out dsaprivatekey.pem -des3 dsaparams.pem
```

Mediante il seguente comando è possibile estrarre la chiave pubblica dal file `dsaprivatekey.pem`

```
openssl dsa -in dsaprivatekey.pem -pubout -out dsapublickey.pem
```

Firma DSA in OpenSSL

(Esempio Generazione Chiavi)

Mediante il seguente comando
utilizzando i parametri

➤ Tale coppia sarà cifrata

➤ Il risultato verrà scritto

Per ottenere la lista
completa delle opzioni è
possibile utilizzare il
comando `man gendsa`

generare una coppia di chiavi DSA,
parametri `dsaparams.pem`

nel file `dsaprivatekey.pem`

```
openssl gendsa -out dsaprivatekey.pem -des3 dsaparams.pem
```

Mediante il seguente comando
dal file `dsaprivatekey.pem`

Per ottenere la lista
completa delle opzioni è
possibile utilizzare il
comando `man dsa`

estrarre la chiave pubblica

```
openssl dsa -in dsaprivatekey.pem -pubout -out dsapublickey.pem
```

Private-Key: (1024 bit)

priv:

1a:97:a0:8b:5d:1c:d8:72:6f:aa:a7:6c:6c:c4:85:
6d:f2:1f:b9:88

pub:

00:8f:8f:f0:a0:7c:7a:bd:e9:27:d0:e7:bb:92:85:
71:af:5b:d2:dc:50:74:1f:d6:41:cb:41:3a:a3:42:
ab:fc:26:f7:ef:e5:38:18:37:5f:17:0e:26:6b:d8:
62:07:ba:a6:2b:23:00:d9:b0:1a:56:22:3f:e1:f6:
88:da:70:d1:1b:34:09:7a:04:c6:35:a1:a4:c6:77:
19:94:6a:29:d1:fc:40:1a:df:4e:4b:dc:a8:b1:4d:
40:a7:f0:51:c0:95:ac:a2:fb:f5:82:74:16:c2:7c:
26:26:b0:3c:e7:09:ed:9a:d6:5e:77:ce:b0:bf:c4:
e4:23:4c:3a:bd:9a:ba:7c:72

P:

00:a0:d5:c6:c6:85:21:8a:fc:a5:90:b8:19:24:4b:
07:11:b6:6c:41:1f:3d:15:71:52:9c:d1:6e:9e:06:
27:1c:e5:fa:d5:90:ba:55:43:de:57:a6:71:58:1f:
08:20:61:9b:31:9e:c9:e8:c9:ba:d5:f6:02:ec:17:
ad:00:fb:55:c3:62:b6:c9:81:8a:68:74:ab:1b:17:
fa:83:37:61:b7:4d:6c:f3:f2:70:20:95:ec:f6:c1:
c9:b0:f0:f4:28:33:62:b4:56:64:9e:66:e0:72:77:
6f:e7:f1:ad:db:b4:8e:94:57:fa:4f:81:6e:69:b1:
d6:7c:ea:e6:84:53:9b:ea:dd

Q:

00:b7:f3:a3:4a:6b:59:b0:06:71:29:f8:d2:30:0f:
cb:c9:63:75:2f:5b

G:

32:5b:6c:78:33:10:fa:0d:44:2b:4b:b9:56:08:cb:
fc:84:1b:11:47:10:63:a8:33:3e:0c:09:12:c8:66:
0b:71:9c:f9:65:8c:4c:36:f3:2d:94:7a:f2:c0:eb:
63:2d:44:cb:08:3e:91:ee:e8:51:12:7b:5a:98:37:
cf:a5:46:0f:b0:63:a6:c8:fe:4c:db:34:d5:61:f3:
31:ee:22:df:72:c1:dc:6b:21:9c:14:e0:9d:cb:1d:
3c:ff:77:6b:d4:fd:54:a1:df:fb:a8:41:23:ae:f2:
5c:c2:b0:43:32:0d:c7:f2:7d:69:3f:6f:e7:72:b8:
f0:1b:88:86:04:9d:53:c4

writing DSA key

-----BEGIN DSA PRIVATE KEY-----

```
MIIBuwIBAAKBgQCg1cbGhSGK/KWQuBkkSwcRtmxBHz0VcVKc0W6eBicc5frVklpV
Q95XpnFYHwggYZsxnsnoybrV9gLSF60A+1XDYrbJgYpodKsbF/qDN2G3TWzz8nAg
lez2wcmw8PQoM2K0VmSeZuByd2/n8a3btI6UV/pPgW5psdZ86uaEU5vq3QIVALfz
o0prWbAGcSn40jAPy8ljds9bAoGAMltseDMQ+g1EK0u5VgjL/IQbEucQY6gzPgWJ
Eshmc3Gc+WMMTDbzLZR68sDrYy1Eyw+ke7oURJ7Wpg3z6VGD7Bjpsj+TNs01WHZ
Me4i33LB3GshnBTgncsdPP93a9T9VKHf+6hBI67yXMKwQzINx/J9aT9v53K48BuI
hgSdU8QCgYEAj4/woHx6vekn00e7koVxr1vS3FB0H9ZBy0E6o0Kr/Cb37+U4GDdf
Fw4ma9hiB7qmKyMA2bAaViI/4faI2nDRGzQJegTGNaGkxncZlGop0fxAGt90S9yo
sU1Ap/BRwJWsov1gnQWwnmJrA85wntmtZed86wv8TKI0w6vZq6fHICFBqXoItD
HNhyb6qnbGzEhW3yH7mI
```

-----END DSA PRIVATE KEY-----

Le chiavi DSA in OpenSSL sono rappresentate secondo lo standard PKCS #8 (RFC5208)

➤ <https://tools.ietf.org/html/rfc5208>

Per visualizzare il contenuto del file `dsaprivatekey.pem` è possibile utilizzare il seguente comando

```
openssl dsa -in dsaprivatekey.pem -text
```

Firma DSA in OpenSSL

(Esempio Firma/Verifica)

Mediante il seguente comando è possibile firmare l'hash SHA1 (DSS1) di **file.txt**

- È utilizzata la chiave privata contenuta in **dsaprivatekey.pem**
- La firma è scritta nel file **dsasign.bin**

```
openssl dgst -dss1 -sign dsaprivatekey.pem -out dsasign.bin file.txt
```

Mediante il seguente comando è possibile verificare la firma del file **file.txt**, contenuta in **dsasign.bin**

- È utilizzata la chiave pubblica contenuta in **dsapublickey.pem**

```
openssl dgst -dss1 -verify dsapublickey.pem -signature dsasign.bin file.txt
```

Firma ECDSA in OpenSSL

(Generazione Chiavi)

- OpenSSL fornisce la variante di DSA basata su curve ellittiche (ECDA)
- È possibile generare una coppia di chiavi ECDSA mediante il comando `ecparam`

Opzioni principali del comando `ecparam`

```
openssl ecparam [options]
```

➤ **options**

- `-inform arg` Formato di input, dove **arg** può essere DER o PEM
- `-outform arg` Formato di output, dove **arg** può essere DER o PEM
- `-in arg` Dove **arg** è il file di input
- `-out arg` Dove **arg** è il file di output
- `-name arg` Tipo di curva ellittica da usare per la generazione dei parametri ECDA
- `-list_curves` Se è specificata questa opzione, verrà stampata la lista di curve ellittiche supportate, ed il comando `ecparam` terminerà
- `-genkey` Questa opzione permette di generare una coppia di chiavi ECDSA, in base al tipo di curva ellittica specificata

Firma ECDSA in OpenSSL

(Generazione Chiavi)

- OpenSSL fornisce la variante di DSA basata su curve ellittiche (ECDA)
- È possibile generare una coppia di chiavi ECDSA mediante il comando `ecparam`

Opzioni principali del comando `ecparam`

```
openssl ecparam [options]
```

➤ **options**

- `-inform arg` Formato di input, dove `arg` può essere DER o PEM
- `-outform arg` Formato di output, dove `arg` può essere DER o PEM
- `-in arg` Dove `arg` è il file di input, completa delle opzioni è possibile utilizzare il comando `man ecparam`
- `-out arg` Dove `arg` è il file di output, generazione dei parametri ECDA
- `-list_curves` Se è specificata questa opzione, verrà stampata la lista di curve ellittiche supportate, ed il comando `ecparam` terminerà
- `-genkey` Questa opzione permette di generare una coppia di chiavi ECDSA, in base al tipo di curva ellittica specificata

Firma ECDSA in OpenSSL

(Esempio Generazione Chiavi)

Mediante il seguente comando è possibile generare una coppia di chiavi ECDSA, utilizzando i parametri generati in base alla curva ellittica **brainpoolP160r1**

```
openssl ecparam -genkey -name brainpoolP160r1 -out  
ecdsa_private.pem
```

Mediante il seguente comando è possibile estrarre la chiave pubblica dal file **ecdsa_private.pem**

```
openssl ec -in ecdsa_private.pem -pubout -out ecdsa_public.pem
```

Firma ECDSA in OpenSSL

(Esempio Generazione Chiavi)

Tipo di curva ellittica utilizzata

`brainpoolP160r1: RFC 5639 curve over a 160 bit prime field`

di chiavi

ECDSA, utilizzando i parametri generati

curva ellittica

`brainpoolP160r1`

```
openssl ecparam -genkey -name brainpoolP160r1 -out  
ecdsa_private.pem
```

Mediante il seguente comando è possibile estrarre la chiave pubblica dal file `ecdsa_private.pem`

```
openssl ec -in ecdsa_private.pem -pubout -out ecdsa_public.pem
```

Firma ECDSA in OpenSSL

(Esempio Generazione Chiavi)

Mediante il seguente comando è possibile generare una coppia di chiavi ECDSA, utilizzando i parametri generati in base alla curva ellittica **brainpoolP160r1**

```
openssl ecparam -genkey -name brainpoolP160r1 -out  
ecdsa_private.pem
```

Mediante il seguente comando è possibile generare la chiave pubblica dal file **ecdsa_private.pem**

Per ottenere la lista completa delle opzioni è possibile utilizzare il comando **man ec**

```
openssl ec -in ecdsa_private.pem -pubout -out ecdsa_public.pem
```

Firma ECDSA in OpenSSL

(Esempio Generazione Chiavi)

È possibile visualizzare il contenuto del file `ecdsa_private.pem` mediante il seguente comando

```
openssl ec -in ecdsa_private.pem -text
```

```
read EC key
Private-Key: (160 bit)
priv:
00:95:c8:38:88:9f:f6:1d:12:ec:64:a7:84:2c:0a:
b5:d7:02:3f:de:76
pub:
04:d4:f6:96:78:a2:9c:5e:f8:48:0e:6c:e6:f9:a7:
2e:ac:2a:d8:0c:4f:26:d1:78:95:e7:75:72:ba:80:
11:95:2b:7f:fa:71:4d:f6:89:fc:d9
ASN1 OID: brainpoolP160r1
```

Le chiavi ECDSA in OpenSSL sono rappresentate secondo l'RFC 3279

➤ <https://www.ietf.org/rfc/rfc3279.txt>

Firma ECDSA in OpenSSL

(Esempio Firma/Verifica)

Mediante il seguente comando è possibile firmare l'hash SHA256 di **File.txt**

- È utilizzata la chiave privata contenuta in **ecdsa_private.pem**
- La firma è scritta nel file **ecdsa_sign.bin**

```
openssl dgst -sha256 -sign ecdsa_private.pem  
File.txt > ecdsa_sign.bin
```

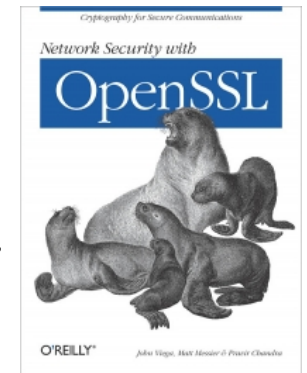
Mediante il seguente comando è possibile verificare la firma del file **File.txt**, contenuta in **ecdsa_sign.bin**

- È utilizzata la chiave pubblica contenuta in **ecdsa_public.pem**

```
openssl dgst -sha256 -verify ecdsa_public.pem  
-signature ecdsa_sign.bin File.txt
```

Bibliografia

- **Network Security with OpenSSL**
Pravir Chandra, Matt Messier and John Viega (2002), O'Reilly
 - Cap. 2.2.1, 2.4.2, 2.4.3
 - Appendix A. Command-Line Reference



- **Documentazione su OpenSSL**
 - <https://www.openssl.org/docs/>

Domande?

