NAME

>   CMS_add0_cert, CMS_add1_cert, CMS_get1_certs, CMS_add0_crl, CMS_add1_crl, CMS_get1_crls, - CMS certificate and CRL utility functions

SYNOPSIS

```
#include <openssl/cms.h>

int CMS_add0_cert(CMS_ContentInfo *cms, X509 *cert);
int CMS_add1_cert(CMS_ContentInfo *cms, X509 *cert);
STACK_OF(X509) *CMS_get1_certs(CMS_ContentInfo *cms);

int CMS_add0_crl(CMS_ContentInfo *cms, X509_CRL *crl);
int CMS_add1_crl(CMS_ContentInfo *cms, X509_CRL *crl);
STACK_OF(X509_CRL) *CMS_get1_crls(CMS_ContentInfo *cms);
```

DESCRIPTION

>   *CMS_add0_cert()* and *CMS_add1_cert()* add certificate **cert** to **cms**. must be of type signed data or enveloped data.

>   *CMS_get1_certs()* returns all certificates in **cms**.

>   *CMS_add0_crl()* and *CMS_add1_crl()* add CRL **crl** to **cms**. *CMS_get1_crls()* returns any CRLs in **cms**.

NOTES

>   The CMS_ContentInfo structure **cms** must be of type signed data or enveloped data or an error will be returned.

>   For signed data certificates and CRLs are added to the **certificates** and **crls** fields of SignedData structure. For enveloped data they are added to **OriginatorInfo**.

>   As the **0** implies *CMS_add0_cert()* adds **cert** internally to **cms** and it must not be freed up after the call as opposed to *CMS_add1_cert()* where **cert** must be freed up.

>   The same certificate or CRL must not be added to the same cms structure more than once.

RETURN VALUES

>   *CMS_add0_cert()*, *CMS_add1_cert()* and *CMS_add0_crl()* and *CMS_add1_crl()* return 1 for success and 0 for failure.

>   *CMS_get1_certs()* and *CMS_get1_crls()* return the STACK of certificates or CRLs or NULL if there are none or an error occurs. The only error which will occur in practice is if the **cms** type is invalid.

SEE ALSO

>   *ERR_get_error(3)* , *CMS_sign(3)* , *CMS_encrypt(3)*

HISTORY

>   *CMS_add0_cert()*, *CMS_add1_cert()*, *CMS_get1_certs()*, *CMS_add0_crl()* and *CMS_get1_crls()* were all first added to OpenSSL 0.9.8