



Full credit is given to the above companies including the OS that this PDF file was generated!

Rocky Enterprise Linux 9.2 Manual Pages on command 'CMS_add0_recipient_key.3ossl'

\$ man CMS_add0_recipient_key.3ossl

CMS_ADD1_RECIPIENT_CERT(3ossl) OpenSSL CMS_ADD1_RECIPIENT_CERT(3ossl)

NAME

CMS_add1_recipient, CMS_add1_recipient_cert, CMS_add0_recipient_key -
add recipients to a CMS enveloped data structure

SYNOPSIS

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

```
CMS_RecipientInfo *CMS_add1_recipient(CMS_ContentInfo *cms, X509 *recip,  
EVP_PKEY *originatorPrivKey,  
X509 *originator, unsigned int flags);
```

```
CMS_RecipientInfo *CMS_add1_recipient_cert(CMS_ContentInfo *cms,  
X509 *recip, unsigned int flags);
```

```
CMS_RecipientInfo *CMS_add0_recipient_key(CMS_ContentInfo *cms, int nid,  
unsigned char *key, size_t keylen,
```

```
unsigned char *id, size_t idlen,  
ASN1_GENERALIZEDTIME *date,  
ASN1_OBJECT *otherTypeid,  
ASN1_TYPE *otherType);
```

DESCRIPTION

`CMS_add1_recipient()` adds recipient `recip` and provides the originator `pkey` originatorPrivKey and originator certificate `originator` to `CMS_ContentInfo`. The originator-related fields are relevant only in case when the `keyAgreement` method of providing of the shared key is in use.

`CMS_add1_recipient_cert()` adds recipient `recip` to `CMS_ContentInfo` enveloped data structure `cms` as a `KeyTransRecipientInfo` structure.

`CMS_add0_recipient_key()` adds symmetric key `key` of length `keylen` using wrapping algorithm `nid`, identifier `id` of length `idlen` and optional values `date`, `otherTypeid` and `otherType` to `CMS_ContentInfo` enveloped data structure `cms` as a `KEKRecipientInfo` structure.

The `CMS_ContentInfo` structure should be obtained from an initial call to `CMS_encrypt()` with the flag `CMS_PARTIAL` set.

NOTES

The main purpose of this function is to provide finer control over a `CMS` enveloped data structure where the simpler `CMS_encrypt()` function defaults are not appropriate. For example if one or more `KEKRecipientInfo` structures need to be added. New attributes can also be added using the returned `CMS_RecipientInfo` structure and the `CMS` attribute utility functions.

OpenSSL will by default identify recipient certificates using issuer name and serial number. If `CMS_USE_KEYID` is set it will use the subject

key identifier value instead. An error occurs if all recipient certificates do not have a subject key identifier extension.

Currently only AES based key wrapping algorithms are supported for nid, specifically: NID_id_aes128_wrap, NID_id_aes192_wrap and NID_id_aes256_wrap. If nid is set to NID_undef then an AES wrap algorithm will be used consistent with keylen.

RETURN VALUES

CMS_add1_recipient_cert() and CMS_add0_recipient_key() return an internal pointer to the CMS_RecipientInfo structure just added or NULL if an error occurs.

SEE ALSO

ERR_get_error(3), CMS_decrypt(3), CMS_final(3),

HISTORY

CMS_add1_recipient_cert and CMS_add0_recipient_key were added in OpenSSL 3.0.

COPYRIGHT

Copyright 2008-2021 The OpenSSL Project Authors. All Rights Reserved.

Licensed under the Apache License 2.0 (the "License"). You may not use this file except in compliance with the License. You can obtain a copy in the file LICENSE in the source distribution or at <https://www.openssl.org/source/license.html>.

3.0.7 2023-07-13 CMS_ADD1_RECIPIENT_CERT(3openssl)