NAME

CMS_SignerInfo_set1_signer_cert, CMS_get0_SignerInfos, CMS_SignerInfo_get0_signer_id, CMS_SignerInfo_get0_signature, CMS_SignerInfo_cert_cmp - CMS signedData signer functions

SYNOPSIS

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

STACK_OF(CMS_SignerInfo) *CMS_get0_SignerInfos(CMS_ContentInfo *cms);

int CMS_SignerInfo_get0_signer_id(CMS_SignerInfo *si, ASN1_OCTET_STRING **keyid,
                                  X509_NAME **issuer, ASN1_INTEGER **sno);
ASN1_OCTET_STRING *CMS_SignerInfo_get0_signature(CMS_SignerInfo *si);
int CMS_SignerInfo_cert_cmp(CMS_SignerInfo *si, X509 *cert);
void CMS_SignerInfo_set1_signer_cert(CMS_SignerInfo *si, X509 *signer);
```

DESCRIPTION

The function **CMS_get0_SignerInfos()** returns all the CMS_SignerInfo structures associated with a CMS signedData structure.

CMS_SignerInfo_get0_signer_id() retrieves the certificate signer identifier associated with a specific CMS_SignerInfo structure **si**. Either the keyidentifier will be set in **keyid** or **both** issuer name and serial number in **issuer** and **sno**.

CMS_SignerInfo_get0_signature() retrieves the signature associated with **si** in a pointer to an ASN1_OCTET_STRING structure. This pointer returned corresponds to the internal signature value if **si** so it may be read or modified.

CMS_SignerInfo_cert_cmp() compares the certificate **cert** against the signer identifier **si**. It returns zero if the comparison is successful and non zero if not.

CMS_SignerInfo_set1_signer_cert() sets the signers certificate of **si** to **signer**.

NOTES

The main purpose of these functions is to enable an application to lookup signers certificates using any appropriate technique when the simpler method of **CMS_verify()** is not appropriate.

In typical usage and application will retrieve all CMS_SignerInfo structures using **CMS_get0_SignerInfo()** and retrieve the identifier information using CMS. It will then obtain the signer certificate by some unspecified means (or return and error if it cannot be found) and set it using **CMS_SignerInfo_set1_signer_cert()**.

Once all signer certificates have been set **CMS_verify()** can be used.

Although **CMS_get0_SignerInfos()** can return NULL if an error occurs **or** if there are no signers this is not a problem in practice because the only error which can occur is if the **cms** structure is not of type signedData due to application error.

RETURN VALUES

CMS_get0_SignerInfos() returns all CMS_SignerInfo structures, or NULL there are no signers or an error occurs.

CMS_SignerInfo_get0_signer_id() returns 1 for success and 0 for failure.

CMS_SignerInfo_cert_cmp() returns 0 for a successful comparison and non zero otherwise.

CMS_SignerInfo_set1_signer_cert() does not return a value.

Any error can be obtained from **ERR_get_error**(3)

SEE ALSO

ERR_get_error(3), **CMS_verify**(3)

COPYRIGHT