



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

Enterprise Linux Release 9.2 Manual Pages on 'OSSL_CRMF_MSG_set1_regCtrl_protocolEncrKey.3oss1' co

```
$ man OSSL_CRMF_MSG_set1_regCtrl_protocolEncrKey.3oss1
```

```
OSSL_CRMF_MSG_SET1_REGCTRL_REGTOKEN(OSSL_CRMF_MSG_SET1_REGCTRL_REGTOKEN(3oss1)
```

NAME

```
OSSL_CRMF_MSG_get0_regCtrl_regToken,  
OSSL_CRMF_MSG_set1_regCtrl_regToken,  
OSSL_CRMF_MSG_get0_regCtrl_authenticator,  
OSSL_CRMF_MSG_set1_regCtrl_authenticator,  
OSSL_CRMF_MSG_PKIPublicationInfo_push0_SinglePubInfo,  
OSSL_CRMF_MSG_set0_SinglePubInfo,  
OSSL_CRMF_MSG_set_PKIPublicationInfo_action,  
OSSL_CRMF_MSG_get0_regCtrl_pkiPublicationInfo,  
OSSL_CRMF_MSG_set1_regCtrl_pkiPublicationInfo,  
OSSL_CRMF_MSG_get0_regCtrl_protocolEncrKey,  
OSSL_CRMF_MSG_set1_regCtrl_protocolEncrKey,  
OSSL_CRMF_MSG_get0_regCtrl_oldCertID,  
OSSL_CRMF_MSG_set1_regCtrl_oldCertID, OSSL_CRMF_CERTID_gen - functions  
getting or setting CRMF Registration Controls
```

SYNOPSIS

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

```
ASN1_UTF8STRING
```

```
*OSSL_CRMF_MSG_get0_regCtrl_regToken(const OSSL_CRMF_MSG *msg);
```

```

int OSSL_CRMF_MSG_set1_regCtrl_regToken(OSSL_CRMF_MSG *msg,
                                         const ASN1_UTF8STRING *tok);

ASN1_UTF8STRING
*OSSL_CRMF_MSG_get0_regCtrl_authenticator(const OSSL_CRMF_MSG *msg);
int OSSL_CRMF_MSG_set1_regCtrl_authenticator(OSSL_CRMF_MSG *msg,
                                             const ASN1_UTF8STRING *auth);

int OSSL_CRMF_MSG_PKIPublicationInfo_push0_SinglePubInfo(
    OSSL_CRMF_PKIPUBLICATIONINFO *pi,
    OSSL_CRMF_SINGLEPUBINFO *spi);
int OSSL_CRMF_MSG_set0_SinglePubInfo(OSSL_CRMF_SINGLEPUBINFO *spi,
                                     int method, GENERAL_NAME *nm);
int OSSL_CRMF_MSG_set_PKIPublicationInfo_action(
    OSSL_CRMF_PKIPUBLICATIONINFO *pi, int action);

OSSL_CRMF_PKIPUBLICATIONINFO
*OSSL_CRMF_MSG_get0_regCtrl_pkiPublicationInfo(const OSSL_CRMF_MSG *msg);
int OSSL_CRMF_MSG_set1_regCtrl_pkiPublicationInfo(OSSL_CRMF_MSG *msg,
                                                  const OSSL_CRMF_PKIPUBLICATIONINFO *pi);

X509_PUBKEY
*OSSL_CRMF_MSG_get0_regCtrl_protocolEncrKey(const OSSL_CRMF_MSG *msg);
int OSSL_CRMF_MSG_set1_regCtrl_protocolEncrKey(OSSL_CRMF_MSG *msg,
                                              const X509_PUBKEY *pubkey);

OSSL_CRMF_CERTID
*OSSL_CRMF_MSG_get0_regCtrl_oldCertID(const OSSL_CRMF_MSG *msg);
int OSSL_CRMF_MSG_set1_regCtrl_oldCertID(OSSL_CRMF_MSG *msg,
                                         const OSSL_CRMF_CERTID *cid);

OSSL_CRMF_CERTID *OSSL_CRMF_CERTID_gen(const X509_NAME *issuer,
                                       const ASN1_INTEGER *serial);

```

DESCRIPTION

Each of the OSSL_CRMF_MSG_get0_regCtrl_X() functions returns the respective control X in the given msg, if present.

OSSL_CRMF_MSG_set1_regCtrl_regToken() sets the regToken control in the

given msg copying the given tok as value. See RFC 4211, section 6.1.

OSSL_CRMF_MSG_set1_regCtrl_authenticator() sets the authenticator control in the given msg copying the given auth as value. See RFC 4211, section 6.2.

OSSL_CRMF_MSG_PKIPublicationInfo_push0_SinglePubInfo() pushes the given spi to si. Consumes the spi pointer.

OSSL_CRMF_MSG_set0_SinglePubInfo() sets in the given SinglePubInfo spi the method and publication location, in the form of a GeneralName, nm.

The publication location is optional, and therefore nm may be NULL.

The function consumes the nm pointer if present. Available methods are:

```
# define OSSL_CRMF_PUB_METHOD_DONTCARE 0
# define OSSL_CRMF_PUB_METHOD_X500    1
# define OSSL_CRMF_PUB_METHOD_WEB     2
# define OSSL_CRMF_PUB_METHOD_LDAP    3
```

OSSL_CRMF_MSG_set_PKIPublicationInfo_action() sets the action in the given pi using the given action as value. See RFC 4211, section 6.3.

Available actions are:

```
# define OSSL_CRMF_PUB_ACTION_DONTPUBLISH 0
# define OSSL_CRMF_PUB_ACTION_PLEASEPUBLISH 1
```

OSSL_CRMF_MSG_set1_regCtrl_pkiPublicationInfo() sets the pkiPublicationInfo control in the given msg copying the given tok as value. See RFC 4211, section 6.3.

OSSL_CRMF_MSG_set1_regCtrl_protocolEncrKey() sets the protocolEncrKey control in the given msg copying the given pubkey as value. See RFC 4211 section 6.6.

OSSL_CRMF_MSG_set1_regCtrl_oldCertID() sets the oldCertID regToken control in the given msg copying the given cid as value. See RFC 4211, section 6.5.

OSSL_CRMF_CERTID_gen produces an OSSL_CRMF_CERTID_gen structure copying the given issuer name and serial number.

RETURN VALUES

All OSSL_CRMF_MSG_get0_*() functions return the respective pointer value or NULL if not present and on error.

All OSSL_CRMF_MSG_set1_*() functions return 1 on success, 0 on error.

OSSL_CRMF_CERTID_gen() returns a pointer to the resulting structure or NULL on error.

NOTES

A function OSSL_CRMF_MSG_set1_regCtrl_pkiArchiveOptions() for setting an Archive Options Control is not yet implemented due to missing features to create the needed OSSL_CRMF_PKIARCHIVEOPTINS content.

SEE ALSO

RFC 4211

HISTORY

The OpenSSL CRMF support was added in OpenSSL 3.0.

COPYRIGHT

Copyright 2007-2022 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

200SSL_CRMF_MSG_SET1_REGCTRL_REGTOKEN(3ossl)