

NAME

OSSL_CMP_ITAV_create, OSSL_CMP_ITAV_set0, OSSL_CMP_ITAV_get0_type, OSSL_CMP_ITAV_get0_value, OSSL_CMP_ITAV_push0_stack_item - OSSL_CMP_ITAV utility functions

SYNOPSIS

```
#include <openssl/cmp.h>
OSSL_CMP_ITAV *OSSL_CMP_ITAV_create(ASN1_OBJECT *type, ASN1_TYPE *value);
void OSSL_CMP_ITAV_set0(OSSL_CMP_ITAV *itav, ASN1_OBJECT *type,
                        ASN1_TYPE *value);
ASN1_OBJECT *OSSL_CMP_ITAV_get0_type(const OSSL_CMP_ITAV *itav);
ASN1_TYPE *OSSL_CMP_ITAV_get0_value(const OSSL_CMP_ITAV *itav);

int OSSL_CMP_ITAV_push0_stack_item(STACK_OF(OSSL_CMP_ITAV) **itav_sk_p,
                                  OSSL_CMP_ITAV *itav);
```

DESCRIPTION

Certificate Management Protocol (CMP, RFC 4210) extension to OpenSSL

ITAV is short for InfoTypeAndValue. This type is defined in RFC 4210 section 5.3.19 and Appendix F. It is used at various places in CMP messages, e.g., in the generalInfo PKIHeader field, to hold a key-value pair.

OSSL_CMP_ITAV_create() creates a new **OSSL_CMP_ITAV** structure and fills it in. It combines **OSSL_CMP_ITAV_new()** and **OSSL_CMP_ITAV_set0()**.

OSSL_CMP_ITAV_set0() sets the *itav* with an infoType of *type* and an infoValue of *value*. This function uses the pointers *type* and *value* internally, so they must **not** be freed up after the call.

OSSL_CMP_ITAV_get0_type() returns a direct pointer to the infoType in the *itav*.

OSSL_CMP_ITAV_get0_value() returns a direct pointer to the infoValue in the *itav* as generic **ASN1_TYPE** pointer.

OSSL_CMP_ITAV_push0_stack_item() pushes *itav* to the stack pointed to by **itav_sk_p*. It creates a new stack if **itav_sk_p* points to NULL.

NOTES

CMP is defined in RFC 4210 (and CRMF in RFC 4211).

RETURN VALUES

OSSL_CMP_ITAV_create() returns a pointer to the ITAV structure on success, or NULL on error.

OSSL_CMP_ITAV_set0() does not return a value.

OSSL_CMP_ITAV_get0_type() and **OSSL_CMP_ITAV_get0_value()** return the respective pointer or NULL if their input is NULL.

OSSL_CMP_ITAV_push0_stack_item() returns 1 on success, 0 on error.

EXAMPLES

The following code creates and sets a structure representing a generic InfoTypeAndValue sequence, using an OID created from text as type, and an integer as value. Afterwards, it is pushed to the **OSSL_CMP_CTX** to be later included in the requests' PKIHeader's genInfo field.

```
ASN1_OBJECT *type = OBJ_txt2obj("1.2.3.4.5", 1);
if (type == NULL) ...

ASN1_INTEGER *asn1int = ASN1_INTEGER_new();
if (asn1int == NULL || !ASN1_INTEGER_set(asn1int, 12345)) ...

ASN1_TYPE *val = ASN1_TYPE_new();
if (val == NULL) ...
ASN1_TYPE_set(val, V_ASN1_INTEGER, asn1int);

OSSL_CMP_ITAV *itav = OSSL_CMP_ITAV_create(type, val);
if (itav == NULL) ...

OSSL_CMP_CTX *ctx = OSSL_CMP_CTX_new();
if (ctx == NULL || !OSSL_CMP_CTX_geninfo_push0_ITAV(ctx, itav)) {
    OSSL_CMP_ITAV_free(itav); /* also frees type and val */
    goto err;
}

...

OSSL_CMP_CTX_free(ctx); /* also frees itav */
```

SEE ALSO

OSSL_CMP_CTX_new(3), **OSSL_CMP_CTX_free(3)**, **ASN1_TYPE_set(3)**

HISTORY

The OpenSSL CMP support was added in OpenSSL 3.0.

COPYRIGHT

Copyright 2007-2021 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>.