## NAME

OCSP_request_add1_nonce, OCSP_basic_add1_nonce, OCSP_check_nonce, OCSP_copy_nonce - OCSP nonce functions

## SYNOPSIS

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

int OCSP_request_add1_nonce(OCSP_REQUEST *req, unsigned char *val, int len);
int OCSP_basic_add1_nonce(OCSP_BASICRESP *resp, unsigned char *val, int len);
int OCSP_copy_nonce(OCSP_BASICRESP *resp, OCSP_REQUEST *req);
int OCSP_check_nonce(OCSP_REQUEST *req, OCSP_BASICRESP *resp);
```

## DESCRIPTION

**OCSP_request_add1_nonce()** adds a nonce of value **val** and length **len** to OCSP request **req**. If **val** is **NULL** a random nonce is used. If **len** is zero or negative a default length will be used (currently 16 bytes).

**OCSP_basic_add1_nonce()** is identical to **OCSP_request_add1_nonce()** except it adds a nonce to OCSP basic response **resp**.

**OCSP_check_nonce()** compares the nonce value in **req** and **resp**.

**OCSP_copy_nonce()** copies any nonce value present in **req** to **resp**.

## RETURN VALUES

**OCSP_request_add1_nonce()** and **OCSP_basic_add1_nonce()** return 1 for success and 0 for failure.

**OCSP_copy_nonce()** returns 1 if a nonce was successfully copied, 2 if no nonce was present in **req** and 0 if an error occurred.

**OCSP_check_nonce()** returns the result of the nonce comparison between **req** and **resp**. The return value indicates the result of the comparison. If nonces are present and equal 1 is returned. If the nonces are absent 2 is returned. If a nonce is present in the response only 3 is returned. If nonces are present and unequal 0 is returned. If the nonce is present in the request only then -1 is returned.

## NOTES

For most purposes the nonce value in a request is set to a random value so the **val** parameter in **OCSP_request_add1_nonce()** is usually NULL.

An OCSP nonce is typically added to an OCSP request to thwart replay attacks by checking the same

nonce value appears in the response.

Some responders may include a nonce in all responses even if one is not supplied.

Some responders cache OCSP responses and do not sign each response for performance reasons. As a result they do not support nonces.

The return values of **OCSP_check_nonce()** can be checked to cover each case. A positive return value effectively indicates success: nonces are both present and match, both absent or present in the response only. A nonzero return additionally covers the case where the nonce is present in the request only: this will happen if the responder doesn't support nonces. A zero return value indicates present and mismatched nonces: this should be treated as an error condition.

## SEE ALSO

**crypto**(7), **OCSP_cert_to_id**(3), **OCSP_REQUEST_new**(3), **OCSP_resp_find_status**(3), **OCSP_response_status**(3), **OCSP_sendreq_new**(3)

## COPYRIGHT