

**NAME**

DSA\_sign, DSA\_sign\_setup, DSA\_verify - DSA signatures

**SYNOPSIS**

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

The following functions have been deprecated since OpenSSL 3.0, and can be hidden entirely by defining **OPENSSL\_API\_COMPAT** with a suitable version value, see **openssl\_user\_macros(7)**:

```
int DSA_sign(int type, const unsigned char *dgst, int len,  
            unsigned char *sigret, unsigned int *siglen, DSA *dsa);
```

```
int DSA_sign_setup(DSA *dsa, BN_CTX *ctx, BIGNUM **kinvp, BIGNUM **r);
```

```
int DSA_verify(int type, const unsigned char *dgst, int len,  
             unsigned char *sigbuf, int siglen, DSA *dsa);
```

**DESCRIPTION**

All of the functions described on this page are deprecated. Applications should instead use **EVP\_PKEY\_sign\_init(3)**, **EVP\_PKEY\_sign(3)**, **EVP\_PKEY\_verify\_init(3)** and **EVP\_PKEY\_verify(3)**.

**DSA\_sign()** computes a digital signature on the **len** byte message digest **dgst** using the private key **dsa** and places its ASN.1 DER encoding at **sigret**. The length of the signature is places in **\*siglen**. **sigret** must point to **DSA\_size(dsa)** bytes of memory.

**DSA\_sign\_setup()** is defined only for backward binary compatibility and should not be used. Since OpenSSL 1.1.0 the DSA type is opaque and the output of **DSA\_sign\_setup()** cannot be used anyway: calling this function will only cause overhead, and does not affect the actual signature (pre-)computation.

**DSA\_verify()** verifies that the signature **sigbuf** of size **siglen** matches a given message digest **dgst** of size **len**. **dsa** is the signer's public key.

The **type** parameter is ignored.

The random generator must be seeded when **DSA\_sign()** (or **DSA\_sign\_setup()**) is called. If the automatic seeding or reseeding of the OpenSSL CSPRNG fails due to external circumstances (see **RAND(7)**), the operation will fail.

**RETURN VALUES**

**DSA\_sign()** and **DSA\_sign\_setup()** return 1 on success, 0 on error. **DSA\_verify()** returns 1 for a valid signature, 0 for an incorrect signature and -1 on error. The error codes can be obtained by **ERR\_get\_error(3)**.

**CONFORMING TO**

US Federal Information Processing Standard FIPS186-4 (Digital Signature Standard, DSS), ANSI X9.30

**SEE ALSO**

**DSA\_new(3)**, **ERR\_get\_error(3)**, **RAND\_bytes(3)**, **DSA\_do\_sign(3)**, **RAND(7)**

**HISTORY**

All of these functions were deprecated in OpenSSL 3.0.

**COPYRIGHT**

Copyright 2000-2022 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>>.