



*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 'EVP\_PKEY\_CTX\_set\_rsa\_pss\_keygen\_mgf1\_md.3ossl' co**

**`$ man EVP_PKEY_CTX_set_rsa_pss_keygen_mgf1_md.3ossl`**

**`EVP_PKEY_CTX_SET_RSA_PSS_KEYGEN_MD(3OEVP_PKEY_CTX_SET_RSA_PSS_KEYGEN_MD(3ossl)`**

## NAME

EVP\_PKEY\_CTX\_set\_rsa\_pss\_keygen\_md,  
EVP\_PKEY\_CTX\_set\_rsa\_pss\_keygen\_md\_name,  
EVP\_PKEY\_CTX\_set\_rsa\_pss\_keygen\_mgf1\_md,  
EVP\_PKEY\_CTX\_set\_rsa\_pss\_keygen\_mgf1\_md\_name,  
EVP\_PKEY\_CTX\_set\_rsa\_pss\_keygen\_saltlen - EVP\_PKEY RSA-PSS algorithm  
support functions

## SYNOPSIS

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

```
int EVP_PKEY_CTX_set_rsa_pss_keygen_md(EVP_PKEY_CTX *pctx,  
    const EVP_MD *md);
```

```
int EVP_PKEY_CTX_set_rsa_pss_keygen_md_name(EVP_PKEY_CTX *ctx,  
    const char *mdname,  
    const char *mdprops);
```

```
int EVP_PKEY_CTX_set_rsa_pss_keygen_mgf1_md(EVP_PKEY_CTX *pctx,  
    const EVP_MD *md);
```

```
int EVP_PKEY_CTX_set_rsa_pss_keygen_mgf1_md_name(EVP_PKEY_CTX *pctx,  
    const char *mdname);
```

```
int EVP_PKEY_CTX_set_rsa_pss_keygen_saltlen(EVP_PKEY_CTX *pctx,
```

```
int saltlen);
```

## DESCRIPTION

These are the functions that implement RSA-PSS(7).

### Signing and Verification

The macro `EVP_PKEY_CTX_set_rsa_padding()` is supported but an error is returned if an attempt is made to set the padding mode to anything other than PSS. It is otherwise similar to the RSA version.

The `EVP_PKEY_CTX_set_rsa_pss_saltlen()` macro is used to set the salt length. If the key has usage restrictions then an error is returned if an attempt is made to set the salt length below the minimum value. It is otherwise similar to the RSA operation except detection of the salt length (using `RSA_PSS_SALTLEN_AUTO`) is not supported for verification if the key has usage restrictions.

The `EVP_PKEY_CTX_set_signature_md(3)` and `EVP_PKEY_CTX_set_rsa_mgf1_md(3)` functions are used to set the digest and MGF1 algorithms respectively. If the key has usage restrictions then an error is returned if an attempt is made to set the digest to anything other than the restricted value. Otherwise these are similar to the RSA versions.

### Key Generation

As with RSA key generation the `EVP_PKEY_CTX_set_rsa_keygen_bits()` and `EVP_PKEY_CTX_set_rsa_keygen_pubexp()` macros are supported for RSA-PSS: they have exactly the same meaning as for the RSA algorithm.

Optional parameter restrictions can be specified when generating a PSS key. If any restrictions are set (using the macros described below) then all parameters are restricted. For example, setting a minimum salt length also restricts the digest and MGF1 algorithms. If any

restrictions are in place then they are reflected in the corresponding parameters of the public key when (for example) a certificate request is signed.

`EVP_PKEY_CTX_set_rsa_pss_keygen_md()` restricts the digest algorithm the generated key can use to md. `EVP_PKEY_CTX_set_rsa_pss_keygen_md_name()` does the same thing, but passes the algorithm by name rather than by `EVP_MD`.

`EVP_PKEY_CTX_set_rsa_pss_keygen_mgf1_md()` restricts the MGF1 algorithm the generated key can use to md.

`EVP_PKEY_CTX_set_rsa_pss_keygen_mgf1_md_name()` does the same thing, but passes the algorithm by name rather than by `EVP_MD`.

`EVP_PKEY_CTX_set_rsa_pss_keygen_saltlen()` restricts the minimum salt length to saltlen.

## NOTES

A context for the RSA-PSS algorithm can be obtained by calling:

```
EVP_PKEY_CTX *pctx = EVP_PKEY_CTX_new_id(EVP_PKEY_RSA_PSS, NULL);
```

## RETURN VALUES

All these functions return 1 for success and 0 or a negative value for failure. In particular a return value of -2 indicates the operation is not supported by the public key algorithm.

## SEE ALSO

`RSA-PSS(7)`, `EVP_PKEY_CTX_new(3)`, `EVP_PKEY_CTX_ctrl_str(3)`,  
`EVP_PKEY_derive(3)`

## COPYRIGHT

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                    202EVP\_PKEY\_CTX\_SET\_RSA\_PSS\_KEYGEN\_MD(3ossl)