NAME

EVP_PKEY−RSA, EVP_KEYMGMT−RSA, RSA − EVP_PKEY RSA keytype and algorithm support

DESCRIPTION

The **RSA** keytype is implemented in OpenSSL's default and FIPS providers. That implementation supports the basic RSA keys, containing the modulus *n*, the public exponent *e*, the private exponent *d*, and a collection of prime factors, exponents and coefficient for CRT calculations, of which the first few are known as *p* and *q*, *dP* and *dQ*, and *qInv*.

Common RSA parameters

In addition to the common parameters that all keytypes should support (see ''Common parameters'' in **provider−keymgmt** (7)), the **RSA** keytype implementation supports the following.

''n'' (**OSSL_PKEY_PARAM_RSA_N**) <unsigned integer>
The RSA ''n'' value.

''e'' (**OSSL_PKEY_PARAM_RSA_E**) <unsigned integer>
The RSA ''e'' value.

''d'' (**OSSL_PKEY_PARAM_RSA_D**) <unsigned integer>
The RSA ''d'' value.

''rsa−factor1'' (**OSSL_PKEY_PARAM_RSA_FACTOR1**) <unsigned integer>
''rsa−factor2'' (**OSSL_PKEY_PARAM_RSA_FACTOR2**) <unsigned integer>
''rsa−factor3'' (**OSSL_PKEY_PARAM_RSA_FACTOR3**) <unsigned integer>
''rsa−factor4'' (**OSSL_PKEY_PARAM_RSA_FACTOR4**) <unsigned integer>
''rsa−factor5'' (**OSSL_PKEY_PARAM_RSA_FACTOR5**) <unsigned integer>
''rsa−factor6'' (**OSSL_PKEY_PARAM_RSA_FACTOR6**) <unsigned integer>
''rsa−factor7'' (**OSSL_PKEY_PARAM_RSA_FACTOR7**) <unsigned integer>
''rsa−factor8'' (**OSSL_PKEY_PARAM_RSA_FACTOR8**) <unsigned integer>
''rsa−factor9'' (**OSSL_PKEY_PARAM_RSA_FACTOR9**) <unsigned integer>
''rsa−factor10'' (**OSSL_PKEY_PARAM_RSA_FACTOR10**) <unsigned integer>
RSA prime factors. The factors are known as ''p'', ''q'' and ''r_i'' in RFC8017. Up to eight additional ''r_i'' prime factors are supported.

''rsa−exponent1'' (**OSSL_PKEY_PARAM_RSA_EXPONENT1**) <unsigned integer>
''rsa−exponent2'' (**OSSL_PKEY_PARAM_RSA_EXPONENT2**) <unsigned integer>
''rsa−exponent3'' (**OSSL_PKEY_PARAM_RSA_EXPONENT3**) <unsigned integer>
''rsa−exponent4'' (**OSSL_PKEY_PARAM_RSA_EXPONENT4**) <unsigned integer>
''rsa−exponent5'' (**OSSL_PKEY_PARAM_RSA_EXPONENT5**) <unsigned integer>
''rsa−exponent6'' (**OSSL_PKEY_PARAM_RSA_EXPONENT6**) <unsigned integer>
''rsa−exponent7'' (**OSSL_PKEY_PARAM_RSA_EXPONENT7**) <unsigned integer>
''rsa−exponent8'' (**OSSL_PKEY_PARAM_RSA_EXPONENT8**) <unsigned integer>
''rsa−exponent9'' (**OSSL_PKEY_PARAM_RSA_EXPONENT9**) <unsigned integer>
''rsa−exponent10'' (**OSSL_PKEY_PARAM_RSA_EXPONENT10**) <unsigned integer>
RSA CRT (Chinese Remainder Theorem) exponents. The exponents are known as ''dP'', ''dQ'' and ''d_i in RFC8017''. Up to eight additional ''d_i'' exponents are supported.

''rsa−coefficient1'' (**OSSL_PKEY_PARAM_RSA_COEFFICIENT1**) <unsigned integer>
''rsa−coefficient2'' (**OSSL_PKEY_PARAM_RSA_COEFFICIENT2**) <unsigned integer>
''rsa−coefficient3'' (**OSSL_PKEY_PARAM_RSA_COEFFICIENT3**) <unsigned integer>
''rsa−coefficient4'' (**OSSL_PKEY_PARAM_RSA_COEFFICIENT4**) <unsigned integer>
''rsa−coefficient5'' (**OSSL_PKEY_PARAM_RSA_COEFFICIENT5**) <unsigned integer>
''rsa−coefficient6'' (**OSSL_PKEY_PARAM_RSA_COEFFICIENT6**) <unsigned integer>
''rsa−coefficient7'' (**OSSL_PKEY_PARAM_RSA_COEFFICIENT7**) <unsigned integer>
''rsa−coefficient8'' (**OSSL_PKEY_PARAM_RSA_COEFFICIENT8**) <unsigned integer>
''rsa−coefficient9'' (**OSSL_PKEY_PARAM_RSA_COEFFICIENT9**) <unsigned integer>
RSA CRT (Chinese Remainder Theorem) coefficients. The coefficients are known as ''qInv'' and ''t_i''. Up to eight additional ''t_i'' exponents are supported.

RSA key generation parameters

When generating RSA keys, the following key generation parameters may be used.

"bits" (**OSSL_PKEY_PARAM_RSA_BITS**) <unsigned integer>

The value should be the cryptographic length for the **RSA** cryptosystem, in bits.

"primes" (**OSSL_PKEY_PARAM_RSA_PRIMES**) <unsigned integer>

The value should be the number of primes for the generated **RSA** key. The default is 2. It isn't permitted to specify a larger number of primes than 10. Additionally, the number of primes is limited by the length of the key being generated so the maximum number could be less. Some providers may only support a value of 2.

"e" (**OSSL_PKEY_PARAM_RSA_E**) <unsigned integer>

The RSA "e" value. The value may be any odd number greater than or equal to 65537. The default value is 65537. For legacy reasons a value of 3 is currently accepted but is deprecated.

RSA key generation parameters for FIPS module testing

When generating RSA keys, the following additional key generation parameters may be used for algorithm testing purposes only. Do not use these to generate RSA keys for a production environment.

"xp" (**OSSL_PKEY_PARAM_RSA_TEST_XP**) <unsigned integer>
"xq" (**OSSL_PKEY_PARAM_RSA_TEST_XQ**) <unsigned integer>

These 2 fields are normally randomly generated and are used to generate "p" and "q".

"xp1" (**OSSL_PKEY_PARAM_RSA_TEST_XP1**) <unsigned integer>
"xp2" (**OSSL_PKEY_PARAM_RSA_TEST_XP2**) <unsigned integer>
"xq1" (**OSSL_PKEY_PARAM_RSA_TEST_XQ1**) <unsigned integer>
"xq2" (**OSSL_PKEY_PARAM_RSA_TEST_XQ2**) <unsigned integer>

These 4 fields are normally randomly generated. The prime factors "p1", "p2", "q1" and "q2" are determined from these values.

RSA key parameters for FIPS module testing

The following intermediate values can be retrieved only if the values specified in "RSA key generation parameters for FIPS module testing" are set. These should not be accessed in a production environment.

"p1" (**OSSL_PKEY_PARAM_RSA_TEST_P1**) <unsigned integer>
"p2" (**OSSL_PKEY_PARAM_RSA_TEST_P2**) <unsigned integer>
"q1" (**OSSL_PKEY_PARAM_RSA_TEST_Q1**) <unsigned integer>
"q2" (**OSSL_PKEY_PARAM_RSA_TEST_Q2**) <unsigned integer>

The auxiliary probable primes.

CONFORMING TO

FIPS186-4

Section B.3.6 Generation of Probable Primes with Conditions Based on Auxiliary Probable Primes

RFC 8017, excluding RSA-PSS and RSA-OAEP

EXAMPLES

An **EVP_PKEY** context can be obtained by calling:

```
EVP_PKEY_CTX *pctx =
    EVP_PKEY_CTX_new_from_name(NULL, "RSA", NULL);
```

An **RSA** key can be generated simply like this:

```
pkey = EVP_RSA_gen(4096);
```

or like this:

```
EVP_PKEY *pkey = NULL;
EVP_PKEY_CTX *pctx =
    EVP_PKEY_CTX_new_from_name(NULL, "RSA", NULL);

EVP_PKEY_keygen_init(pctx);
```

```
EVP_PKEY_generate(pctx, &pkey);
EVP_PKEY_CTX_free(pctx);
```

An **RSA** key can be generated with key generation parameters:

```
unsigned int primes = 3;
unsigned int bits = 4096;
OSSL_PARAM params[3];
EVP_PKEY *pkey = NULL;
EVP_PKEY_CTX *pctx = EVP_PKEY_CTX_new_from_name(NULL, "RSA", NULL);

EVP_PKEY_keygen_init(pctx);

params[0] = OSSL_PARAM_construct_uint("bits", &bits);
params[1] = OSSL_PARAM_construct_uint("primes", &primes);
params[2] = OSSL_PARAM_construct_end();
EVP_PKEY_CTX_set_params(pctx, params);

EVP_PKEY_generate(pctx, &pkey);
EVP_PKEY_print_private(bio_out, pkey, 0, NULL);
EVP_PKEY_CTX_free(pctx);
```

SEE ALSO

EVP_RSA_gen (3), **EVP_KEYMGMT** (3), **EVP_PKEY** (3), **provider−keymgmt** (7)

COPYRIGHT