NAME

EVP_PKEY−X25519, EVP_PKEY−X448, EVP_PKEY−ED25519, EVP_PKEY−ED448, EVP_KEYMGMT−X25519, EVP_KEYMGMT−X448, EVP_KEYMGMT−ED25519, EVP_KEYMGMT−ED448 − EVP_PKEY X25519, X448, ED25519 and ED448 keytype and algorithm support

DESCRIPTION

The **X25519**, **X448**, **ED25519** and **ED448** keytypes are implemented in OpenSSL's default and FIPS providers. These implementations support the associated key, containing the public key *pub* and the private key *priv*.

No additional parameters can be set during key generation.

Common X25519, X448, ED25519 and ED448 parameters

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

''group'' (**OSSL_PKEY_PARAM_GROUP_NAME**) <UTF8 string>
: This is only supported by X25519 and X448. The group name must be ''x25519'' or ''x448'' respectively for those algorithms. This is only present for consistency with other key exchange algorithms and is typically not needed.

''pub'' (**OSSL_PKEY_PARAM_PUB_KEY**) <octet string>
: The public key value.

''priv'' (**OSSL_PKEY_PARAM_PRIV_KEY**) <octet string>
: The private key value.

''encoded-pub-key'' (**OSSL_PKEY_PARAM_ENCODED_PUBLIC_KEY**) <octet string>
: Used for getting and setting the encoding of a public key for the **X25519** and **X448** key types. Public keys are expected be encoded in a format as defined by RFC7748.

ED25519 and ED448 parameters

''mandatory-digest'' (**OSSL_PKEY_PARAM_MANDATORY_DIGEST**) <UTF8 string>
: The empty string, signifying that no digest may be specified.

CONFORMING TO

RFC 8032
RFC 8410

EXAMPLES

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

```
EVP_PKEY_CTX *pctx =
    EVP_PKEY_CTX_new_from_name(NULL, "X25519", NULL);

EVP_PKEY_CTX *pctx =
    EVP_PKEY_CTX_new_from_name(NULL, "X448", NULL);

EVP_PKEY_CTX *pctx =
    EVP_PKEY_CTX_new_from_name(NULL, "ED25519", NULL);

EVP_PKEY_CTX *pctx =
    EVP_PKEY_CTX_new_from_name(NULL, "ED448", NULL);
```

An **X25519** key can be generated like this:

```
pkey = EVP_PKEY_Q_keygen(NULL, NULL, "X25519");
```

An **X448**, **ED25519**, or **ED448** key can be generated likewise.

SEE ALSO

EVP_KEYMGMT (3), **EVP_PKEY** (3), **provider−keymgmt** (7), **EVP_KEYEXCH−X25519** (7), **EVP_KEYEXCH−X448** (7), **EVP_SIGNATURE−ED25519** (7), **EVP_SIGNATURE−ED448** (7)

COPYRIGHT