

**NAME**

EC\_GFp\_simple\_method, EC\_GFp\_mont\_method, EC\_GFp\_nist\_method,  
 EC\_GFp\_nistp224\_method, EC\_GFp\_nistp256\_method, EC\_GFp\_nistp521\_method,  
 EC\_GF2m\_simple\_method, EC\_METHOD\_get\_field\_type - Functions for obtaining  
 EC\_METHOD objects.

**SYNOPSIS**

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

const EC_METHOD *EC_GFp_simple_method(void);
const EC_METHOD *EC_GFp_mont_method(void);
const EC_METHOD *EC_GFp_nist_method(void);
const EC_METHOD *EC_GFp_nistp224_method(void);
const EC_METHOD *EC_GFp_nistp256_method(void);
const EC_METHOD *EC_GFp_nistp521_method(void);

const EC_METHOD *EC_GF2m_simple_method(void);

int EC_METHOD_get_field_type(const EC_METHOD *meth);
```

**DESCRIPTION**

The Elliptic Curve library provides a number of different implementations through a single common interface. When constructing a curve using `EC_GROUP_new` (see [EC\\_GROUP\\_new\(3\)](#)) an implementation method must be provided. The functions described here all return a const pointer to an `EC_METHOD` structure that can be passed to `EC_GROUP_NEW`. It is important that the correct implementation type for the form of curve selected is used.

For  $F_2^m$  curves there is only one implementation choice, i.e. `EC_GF2_simple_method`.

For  $F_p$  curves the lowest common denominator implementation is the `EC_GFp_simple_method` implementation. All other implementations are based on this one. `EC_GFp_mont_method` builds on `EC_GFp_simple_method` but adds the use of montgomery multiplication (see [BN\\_mod\\_mul\\_montgomery\(3\)](#)). `EC_GFp_nist_method` offers an implementation optimised for use with NIST recommended curves (NIST curves are available through `EC_GROUP_new_by_curve_name` as described in [EC\\_GROUP\\_new\(3\)](#)).

The functions `EC_GFp_nistp224_method`, `EC_GFp_nistp256_method` and `EC_GFp_nistp521_method` offer 64 bit optimised implementations for the NIST P224, P256 and P521 curves respectively. Note, however, that these implementations are not available on all platforms.

`EC_METHOD_get_field_type` identifies what type of field the `EC_METHOD` structure supports, which will be either  $F_2^m$  or  $F_p$ . If the field type is  $F_p$  then the value `NID_X9_62_prime_field` is returned. If the field type is  $F_2^m$  then the value `NID_X9_62_characteristic_two_field` is returned. These values are defined in the `obj_mac.h` header file.

**RETURN VALUES**

All `EC_GFp*` functions and `EC_GF2m_simple_method` always return a const pointer to an `EC_METHOD` structure.

`EC_METHOD_get_field_type` returns an integer that identifies the type of field the `EC_METHOD` structure supports.

**SEE ALSO**

[crypto\(3\)](#), [ec\(3\)](#), [EC\\_GROUP\\_new\(3\)](#), [EC\\_GROUP\\_copy\(3\)](#), [EC\\_POINT\\_new\(3\)](#),  
[EC\\_POINT\\_add\(3\)](#), [EC\\_KEY\\_new\(3\)](#), [d2i\\_ECPKParameters\(3\)](#), [BN\\_mod\\_mul\\_montgomery\(3\)](#)