

NAME

ldns_key_set_algorithm, ldns_key_set_rsa_key, ldns_key_set_dsa_key, ldns_key_set_hmac_key, ldns_key_set_origttl, ldns_key_set_inception, ldns_key_set_expiration, ldns_key_set_pubkey_owner, ldns_key_set_keytag, ldns_key_set_flags, ldns_key_list_set_key_count, ldns_key_algo_supported - set ldns_key attributes

SYNOPSIS

```
#include <stdint.h>
```

```
#include <stdbool.h>
```

```
#include <ldns/ldns.h>
```

```
void ldns_key_set_algorithm(ldns_key *k, ldns_signing_algorithm l);
```

```
void ldns_key_set_rsa_key(ldns_key *k, RSA *r);
```

```
void ldns_key_set_dsa_key(ldns_key *k, DSA *d);
```

```
void ldns_key_set_hmac_key(ldns_key *k, unsigned char *hmac);
```

```
void ldns_key_set_origttl(ldns_key *k, uint32_t t);
```

```
void ldns_key_set_inception(ldns_key *k, uint32_t i);
```

```
void ldns_key_set_expiration(ldns_key *k, uint32_t e);
```

```
void ldns_key_set_pubkey_owner(ldns_key *k, ldns_rdf *r);
```

```
void ldns_key_set_keytag(ldns_key *k, uint16_t tag);
```

```
void ldns_key_set_flags(ldns_key *k, uint16_t flags);
```

```
void ldns_key_list_set_key_count(ldns_key_list *key, size_t count);
```

```
int ldns_key_algo_supported(int algo);
```

DESCRIPTION

ldns_key_set_algorithm() Set the key's algorithm

k: the key

l: the algorithm

ldns_key_set_rsa_key() Set the key's rsa data. The rsa data should be freed by the user.

k: the key

r: the rsa data

ldns_key_set_dsa_key() Set the key's dsa data The dsa data should be freed by the user.

k: the key

d: the dsa data

ldns_key_set_hmac_key() Set the key's hmac data

k: the key

hmac: the raw key data

ldns_key_set_origttl() Set the key's original ttl

k: the key

t: the ttl

ldns_key_set_inception() Set the key's inception date (seconds after epoch)

k: the key

i: the inception

ldns_key_set_expiration() Set the key's expiration date (seconds after epoch)

k: the key

e: the expiration

ldns_key_set_pubkey_owner() Set the key's pubkey owner

k: the key

r: the owner

ldns_key_set_keytag() Set the key's key tag

k: the key

tag: the keytag

ldns_key_set_flags() Set the key's flags

k: the key

flags: the flags

ldns_key_list_set_key_count() Set the keylist's key count to count

key: the key

count: the count

ldns_key_algo_supported() See if a key algorithm is supported

algo: the signing algorithm number.

Returns s true if supported.

AUTHOR

The ldns team at NLnet Labs.

REPORTING BUGS

Please report bugs to ldns-team@nlnetlabs.nl or in our bugzilla at <http://www.nlnetlabs.nl/bugs/index.html>

COPYRIGHT

Copyright (c) 2004 - 2006 NLnet Labs.

Licensed under the BSD License. There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

SEE ALSO

ldns_key_push_key, *ldns_key*. And **perldoc Net::DNS, RFC1034, RFC1035, RFC4033, RFC4034** and **RFC4035**.

REMARKS

This manpage was automatically generated from the ldns source code.