

**NAME**

ldns\_buffer\_write\_at, ldns\_buffer\_write, ldns\_buffer\_write\_string\_at, ldns\_buffer\_write\_string, ldns\_buffer\_write\_u8\_at, ldns\_buffer\_write\_u8, ldns\_buffer\_write\_u16\_at, ldns\_buffer\_write\_u16, ldns\_buffer\_read\_at, ldns\_buffer\_read, ldns\_buffer\_read\_u8\_at, ldns\_buffer\_read\_u8, ldns\_buffer\_read\_u16\_at, ldns\_buffer\_read\_u16, ldns\_buffer\_read\_u32\_at, ldns\_buffer\_read\_u32, ldns\_buffer\_write\_u32, ldns\_buffer\_write\_u32\_at - reading and writing buffers

**SYNOPSIS**

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

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

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

```
void ldns_buffer_write_at(ldns_buffer *buffer, size_t at, const void *data, size_t count);
```

```
void ldns_buffer_write(ldns_buffer *buffer, const void *data, size_t count);
```

```
void ldns_buffer_write_string_at(ldns_buffer *buffer, size_t at, const char *str);
```

```
void ldns_buffer_write_string(ldns_buffer *buffer, const char *str);
```

```
void ldns_buffer_write_u8_at(ldns_buffer *buffer, size_t at, uint8_t data);
```

```
void ldns_buffer_write_u8(ldns_buffer *buffer, uint8_t data);
```

```
void ldns_buffer_write_u16_at(ldns_buffer *buffer, size_t at, uint16_t data);
```

```
void ldns_buffer_write_u16(ldns_buffer *buffer, uint16_t data);
```

```
void ldns_buffer_read_at(const ldns_buffer *buffer, size_t at, void *data, size_t count);
```

```
void ldns_buffer_read(ldns_buffer *buffer, void *data, size_t count);
```

```
uint8_t ldns_buffer_read_u8_at(const ldns_buffer *buffer, size_t at);
```

```
uint8_t ldns_buffer_read_u8(ldns_buffer *buffer);
```

```
uint16_t ldns_buffer_read_u16_at(ldns_buffer *buffer, size_t at);
```

```
uint16_t ldns_buffer_read_u16(ldns_buffer *buffer);

uint32_t ldns_buffer_read_u32_at(ldns_buffer *buffer, size_t at);

uint32_t ldns_buffer_read_u32(ldns_buffer *buffer);

void ldns_buffer_write_u32(ldns_buffer *buffer, uint32_t data);

void ldns_buffer_write_u32_at(ldns_buffer *buffer, size_t at, uint32_t data);
```

## DESCRIPTION

*ldns\_buffer\_write\_at()* writes the given data to the buffer at the specified position

**buffer:** the buffer  
**at:** the position (in number of bytes) to write the data at  
**data:** pointer to the data to write to the buffer  
**count:** the number of bytes of data to write

*ldns\_buffer\_write()* writes count bytes of data to the current position of the buffer

**buffer:** the buffer  
**data:** the data to write  
**count:** the length of the data to write

*ldns\_buffer\_write\_string\_at()* copies the given (null-delimited) string to the specified position at the buffer

**buffer:** the buffer  
**at:** the position in the buffer  
**str:** the string to write

*ldns\_buffer\_write\_string()* copies the given (null-delimited) string to the current position at the buffer

**buffer:** the buffer  
**str:** the string to write

*ldns\_buffer\_write\_u8\_at()* writes the given byte of data at the given position in the buffer

**buffer:** the buffer  
**at:** the position in the buffer  
**data:** the 8 bits to write

*ldns\_buffer\_write\_u8()* writes the given byte of data at the current position in the buffer

**buffer:** the buffer  
**data:** the 8 bits to write

*ldns\_buffer\_write\_u16\_at()* writes the given 2 byte integer at the given position in the buffer

**buffer:** the buffer

**at:** the position in the buffer

**data:** the 16 bits to write

*ldns\_buffer\_write\_u16()* writes the given 2 byte integer at the current position in the buffer

**buffer:** the buffer

**data:** the 16 bits to write

*ldns\_buffer\_read\_at()* copies count bytes of data at the given position to the given data-array

**buffer:** the buffer

**at:** the position in the buffer to start

**data:** buffer to copy to

**count:** the length of the data to copy

*ldns\_buffer\_read()* copies count bytes of data at the current position to the given data-array

**buffer:** the buffer

**data:** buffer to copy to

**count:** the length of the data to copy

*ldns\_buffer\_read\_u8\_at()* returns the byte value at the given position in the buffer

**buffer:** the buffer

**at:** the position in the buffer

Returns 1 byte integer

*ldns\_buffer\_read\_u8()* returns the byte value at the current position in the buffer

**buffer:** the buffer

Returns 1 byte integer

*ldns\_buffer\_read\_u16\_at()* returns the 2-byte integer value at the given position in the buffer

**buffer:** the buffer

**at:** position in the buffer

Returns 2 byte integer

*ldns\_buffer\_read\_u16()* returns the 2-byte integer value at the current position in the buffer

**buffer:** the buffer

Returns 2 byte integer

*ldns\_buffer\_read\_u32\_at()* returns the 4-byte integer value at the given position in the buffer

**buffer:** the buffer

**at:** position in the buffer  
Returns 4 byte integer

*ldns\_buffer\_read\_u32()* returns the 4-byte integer value at the current position in the buffer  
**buffer:** the buffer  
Returns 4 byte integer

*ldns\_buffer\_write\_u32()* writes the given 4 byte integer at the current position in the buffer  
**buffer:** the buffer  
**data:** the 32 bits to write

*ldns\_buffer\_write\_u32\_at()* writes the given 4 byte integer at the given position in the buffer  
**buffer:** the buffer  
**at:** the position in the buffer  
**data:** the 32 bits to write

## AUTHOR

The ldns team at NLnet Labs.

## REPORTING BUGS

Please report bugs to [ldns-team@nlnetlabs.nl](mailto: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\_buffer*. And **perldoc Net::DNS**, **RFC1034**, **RFC1035**, **RFC4033**, **RFC4034** and **RFC4035**.

## REMARKS

This manpage was automatically generated from the ldns source code.