

**NAME**

ldns\_rdf2buffer\_str\_a, ldns\_rdf2buffer\_str\_aaaa, ldns\_rdf2buffer\_str\_str, ldns\_rdf2buffer\_str\_b64, ldns\_rdf2buffer\_str\_hex, ldns\_rdf2buffer\_str\_type, ldns\_rdf2buffer\_str\_class, ldns\_rdf2buffer\_str\_alg, ldns\_rdf2buffer\_str\_loc, ldns\_rdf2buffer\_str\_unknown, ldns\_rdf2buffer\_str\_nsap, ldns\_rdf2buffer\_str\_wks, ldns\_rdf2buffer\_str\_nsec, ldns\_rdf2buffer\_str\_period, ldns\_rdf2buffer\_str\_tsigtime, ldns\_rdf2buffer\_str\_apl, ldns\_rdf2buffer\_str\_int16\_data, ldns\_rdf2buffer\_str\_int16, ldns\_rdf2buffer\_str\_ipseckey - lower level to string conversion functions

**SYNOPSIS**

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

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

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

```
ldns_status ldns_rdf2buffer_str_a(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_aaaa(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_str(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_b64(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_hex(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_type(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_class(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_alg(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_loc(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_unknown(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_nsap(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_wks(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_nsec(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_period(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_tsigtime(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_apl(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_int16_data(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_int16(ldns_buffer *output, const ldns_rdf *rdf);
```

```
ldns_status ldns_rdf2buffer_str_ipseckey(ldns_buffer *output, const ldns_rdf *rdf);
```

## DESCRIPTION

*ldns\_rdf2buffer\_str\_a()* Converts an LDNS\_RDF\_TYPE\_A rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert

**\*output:** The buffer to add the data to

Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_aaaa()* Converts an LDNS\_RDF\_TYPE\_AAAA rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert

**\*output:** The buffer to add the data to

Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_str()* Converts an LDNS\_RDF\_TYPE\_STR rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert

**\*output:** The buffer to add the data to

Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_b64()* Converts an LDNS\_RDF\_TYPE\_B64 rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert

**\*output:** The buffer to add the data to

Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_hex()* Converts an LDNS\_RDF\_TYPE\_HEX rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert

**\*output:** The buffer to add the data to  
Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_type()* Converts an LDNS\_RDF\_TYPE\_TYPE rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert  
**\*output:** The buffer to add the data to  
Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_class()* Converts an LDNS\_RDF\_TYPE\_CLASS rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert  
**\*output:** The buffer to add the data to  
Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_alg()* Converts an LDNS\_RDF\_TYPE\_ALG rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert  
**\*output:** The buffer to add the data to  
Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_loc()* Converts an LDNS\_RDF\_TYPE\_LOC rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert  
**\*output:** The buffer to add the data to  
Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_unknown()* Converts an LDNS\_RDF\_TYPE\_UNKNOWN rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert  
**\*output:** The buffer to add the data to  
Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_nsap()* Converts an LDNS\_RDF\_TYPE\_NSAP rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert  
**\*output:** The buffer to add the data to  
Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_wks()* Converts an LDNS\_RDF\_TYPE\_WKS rdata element to string format and

adds it to the output buffer

**\*rdf:** The rdata to convert

**\*output:** The buffer to add the data to

Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_nsec()* Converts an LDNS\_RDF\_TYPE\_NSEC rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert

**\*output:** The buffer to add the data to

Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_period()* Converts an LDNS\_RDF\_TYPE\_PERIOD rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert

**\*output:** The buffer to add the data to

Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_tsigtime()* Converts an LDNS\_RDF\_TYPE\_TSIGTIME rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert

**\*output:** The buffer to add the data to

Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_apl()* Converts an LDNS\_RDF\_TYPE\_APL rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert

**\*output:** The buffer to add the data to

Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_int16\_data()* Converts an LDNS\_RDF\_TYPE\_INT16\_DATA rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert

**\*output:** The buffer to add the data to

Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_int16()* Converts an LDNS\_RDF\_TYPE\_INT16 rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert

**\*output:** The buffer to add the data to

Returns LDNS\_STATUS\_OK on success, and error status on failure

*ldns\_rdf2buffer\_str\_ipseckey()* Converts an LDNS\_RDF\_TYPE\_IPSECKEY rdata element to string format and adds it to the output buffer

**\*rdf:** The rdata to convert

**\*output:** The buffer to add the data to

Returns LDNS\_STATUS\_OK on success, and error status on failure

## AUTHOR

The ldns team at NLnet Labs.

## REPORTING BUGS

Please report bugs to [dns-team@nlnetlabs.nl](mailto:dns-team@nlnetlabs.nl) or on GitHub at <https://github.com/NLnetLabs/ldns/issues>

## 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

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

## REMARKS

This manpage was automatically generated from the ldns source code.