

NAME

ares_dns_class_fromstr, ares_dns_class_tostr, ares_dns_datatype_t, ares_dns_opcode_tostr, ares_dns_opt_datatype_t, ares_dns_opt_get_datatype, ares_dns_opt_get_name, ares_dns_rcode_tostr, ares_dns_rec_type_fromstr, ares_dns_rr_get_keys, ares_dns_rr_key_datatype, ares_dns_rr_key_to_rec_type, ares_dns_rr_key_tostr, ares_dns_section_tostr, ares_opt_param_t, ares_svcb_param_t – Helper functions for converting dns record identifiers to and from their respective types, as well identifying datatypes for various records.

SYNOPSIS

```
#include <ares.h>
```

```
const char *ares_dns_rec_type_tostr(ares_dns_rec_type_t type);

const char *ares_dns_class_tostr(ares_dns_class_t qclass);

const char *ares_dns_opcode_tostr(ares_dns_opcode_t opcode);

const char *ares_dns_rr_key_tostr(ares_dns_rr_key_t key);

const char *ares_dns_section_tostr(ares_dns_section_t section);

const char *ares_dns_rcode_tostr(ares_dns_rcode_t rcode);

ares_bool_t ares_dns_class_fromstr(ares_dns_class_t *qclass, const char *str);

ares_bool_t ares_dns_rec_type_fromstr(ares_dns_rec_type_t *qtype,
                                     const char *str);

const ares_dns_rr_key_t *ares_dns_rr_get_keys(ares_dns_rec_type_t type,
                                              size_t *cnt);

ares_dns_datatype_t ares_dns_rr_key_datatype(ares_dns_rr_key_t key);

ares_dns_rec_type_t ares_dns_rr_key_to_rec_type(ares_dns_rr_key_t key);

ares_dns_opt_datatype_t ares_dns_opt_get_datatype(ares_dns_rr_key_t key,
                                                  unsigned short opt);

const char *ares_dns_opt_get_name(ares_dns_rr_key_t key, unsigned short opt);
```

ENUMERATIONS

ares_dns_datatype_t - Data types associated with *ares_dns_rr_key_t*:

ARES_DATATYPE_INADDR - IPv4 address as *struct in_addr* *. Use *ares_dns_rr_set_addr(3)* to set and *ares_dns_rr_get_addr(3)* to get.

ARES_DATATYPE_INADDR6 - IPv6 address as *struct ares_in6_addr* *. Use *ares_dns_rr_set_addr6(3)* to set and *ares_dns_rr_get_addr6(3)* to get.

ARES_DATATYPE_U8 - 8bit unsigned integer. Use *ares_dns_rr_set_u8(3)* to set and *ares_dns_rr_get_u8(3)* to get.

ARES_DATATYPE_U16 - 16bit unsigned integer. Use *ares_dns_rr_set_u16(3)* to set and *ares_dns_rr_get_u16(3)* to get.

ARES_DATATYPE_U32 - 32bit unsigned integer. Use *ares_dns_rr_set_u32(3)* to set and *ares_dns_rr_get_u32(3)* to get.

ARES_DATATYPE_NAME - Null-terminated string of a domain name (series of labels). Use *ares_dns_rr_set_str(3)* to set and *ares_dns_rr_get_str(3)* to get.

ARES_DATATYPE_STR - Null-terminated string. Use *ares_dns_rr_set_str(3)* to set and

ares_dns_rr_get_str(3) to get.

ARES_DATATYPE_BIN - Binary Data. Use *ares_dns_rr_set_bin(3)* to set and *ares_dns_rr_get_bin(3)* to get.

ARES_DATATYPE_BINP - Officially defined as binary data, but likely printable. Guaranteed to have a NULL terminator for convenience (not included in length). Use *ares_dns_rr_set_bin(3)* to set and *ares_dns_rr_get_bin(3)* to get.

ARES_DATATYPE_OPT - Array of options. 16bit identifier, Binary data. Use *ares_dns_rr_set_opt(3)* to set and *ares_dns_rr_get_opt(3)* to get.

ARES_DATATYPE_ABINP - Array of binary data, but likely printable. Guaranteed to have a NULL terminator for convenience (not included in length). Use *ares_dns_rr_add_abin(3)* to set and *ares_dns_rr_get_abin(3)* to get. Can also use *ares_dns_rr_set_bin(3)* and *ares_dns_rr_get_bin(3)* if only concatenated strings are desired.

ares_dns_opt_datatype_t - Data types associated with **ARES_DATATYPE_OPT** parameters if known as returned by *ares_dns_opt_get_datatype(3)*:

ARES_OPT_DATATYPE_NONE - No value allowed for this parameter

ARES_OPT_DATATYPE_STR_LIST - List of strings, each prefixed with a single octet representing the length as defined by RFC 1035. Can use *ares_expand_string(3)* until buffer is consumed.

ARES_OPT_DATATYPE_U8_LIST - List of 8bit unsigned integers, concatenated

ARES_OPT_DATATYPE_U16 - 16bit unsigned integer in network byte order

ARES_OPT_DATATYPE_U16_LIST - list of 16bit unsigned integers in network byte order, concatenated.

ARES_OPT_DATATYPE_U32 - 32bit unsigned integer in network byte order

ARES_OPT_DATATYPE_U32_LIST - list of 16bit unsigned integers in network byte order, concatenated.

ARES_OPT_DATATYPE_INADDR4_LIST - List of ipv4 addresses in network byte order, concatenated

ARES_OPT_DATATYPE_INADDR6_LIST - List of ipv6 addresses in network byte order, concatenated

ARES_OPT_DATATYPE_BIN - Binary Data

ARES_OPT_DATATYPE_NAME - DNS Domain Name binary format as defined in RFC1035, can use *ares_expand_name(3)*

ares_svcb_param_t - SVCB (and HTTPS) RR known parameters as returned by *ares_dns_opt_get_datatype(3)* with **ARES_RR_SVCB_PARAMS** or **ARES_RR_HTTPS_PARAMS**:

ARES_SVCB_PARAM_MANDATORY - Mandatory keys in this RR (RFC 9460 Section 8). Datatype: **ARES_OPT_DATATYPE_U16_LIST**

ARES_SVCB_PARAM_ALPN - Additional supported protocols (RFC 9460 Section 7.1). Datatype: **ARES_OPT_DATATYPE_STR_LIST**

ARES_SVCB_PARAM_NO_DEFAULT_ALPN - No support for default protocol (RFC 9460 Section 7.1). Datatype: **ARES_OPT_DATATYPE_NONE**

ARES_SVCB_PARAM_PORT - Port for alternative endpoint (RFC 9460 Section 7.2). Datatype: **ARES_OPT_DATATYPE_U16**

ARES_SVCB_PARAM_IPV4HINT - IPv4 address hints (RFC 9460 Section 7.3). Datatype: **ARES_OPT_DATATYPE_INADDR4_LIST**

ARES_SVCB_PARAM_ECH - RESERVED (held for Encrypted ClientHello)

ARES_SVCB_PARAM_IPV6HINT - IPv6 address hints (RFC 9460 Section 7.3). Datatype: **ARES_OPT_DATATYPE_INADDR6_LIST**

ares_opt_param_t - OPT RR known parameters as returned by *ares_dns_opt_get_datatype(3)* with **ARES_RR_OPT_OPTIONS**:

ARES_OPT_PARAM_LLQ - RFC 8764. Apple's DNS Long-Lived Queries Protocol. Datatype: **ARES_OPT_DATATYPE_BIN**

ARES_OPT_PARAM_UL - <http://files.dns-sd.org/draft-sekar-dns-ul.txt>: Update Lease. Datatype: *ARES_OPT_DATATYPE_U32*

ARES_OPT_PARAM_NSID - RFC 5001. Name Server Identification. Datatype: *ARES_OPT_DATATYPE_BIN*

ARES_OPT_PARAM_DAU - RFC 6975. DNSSEC Algorithm Understood. Datatype: *ARES_OPT_DATATYPE_U8_LIST*

ARES_OPT_PARAM_DHU - RFC 6975. DS Hash Understood. Datatype: *ARES_OPT_DATATYPE_U8_LIST*

ARES_OPT_PARAM_N3U - RFC 6975. NSEC3 Hash Understood. Datatype: *ARES_OPT_DATATYPE_U8_LIST*

ARES_OPT_PARAM_EDNS_CLIENT_SUBNET - RFC 7871. Client Subnet. Datatype: *ARES_OPT_DATATYPE_BIN*

ARES_OPT_PARAM_EDNS_EXPIRE - RFC 7314. Expire Timer. Datatype: *ARES_OPT_DATATYPE_U32*

ARES_OPT_PARAM_COOKIE - RFC 7873. Client and Server Cookies. Datatype: *ARES_OPT_DATATYPE_BIN*

ARES_OPT_PARAM_EDNS_TCP_KEEPALIVE - RFC 7828. TCP Keepalive timeout. Datatype: *ARES_OPT_DATATYPE_U16*

ARES_OPT_PARAM_PADDING - RFC 7830. Padding. Datatype: *ARES_OPT_DATATYPE_BIN*

ARES_OPT_PARAM_CHAIN - RFC 7901. Chain query requests. Datatype: *ARES_OPT_DATATYPE_NAME*

ARES_OPT_PARAM_EDNS_KEY_TAG - RFC 8145. Signaling Trust Anchor Knowledge in DNSSEC. Datatype: *ARES_OPT_DATATYPE_U16_LIST*

ARES_OPT_PARAM_EXTENDED_DNS_ERROR - RFC 8914. Extended ERROR code and message. Datatype: *ARES_OPT_DATATYPE_BIN*

DESCRIPTION

The *ares_dns_rec_type_tostr(3)* function outputs a human readable DNS record type from its numeric form passed in *type*.

The *ares_dns_class_tostr(3)* function outputs a human readable DNS class from its numeric form passed in *qclass*.

The *ares_dns_opcode_tostr(3)* function outputs a human readable DNS opcode from its numeric form in *opcode*.

The *ares_dns_rr_key_tostr(3)* function outputs a human readable DNS Resource Record parameter name from its numeric form in *key*.

The *ares_dns_section_tostr(3)* function outputs a human readable DNS message section from its numeric form in *section*.

The *ares_dns_rcode_tostr(3)* function outputs a human readable DNS response code from its numeric form in *rcode*.

The *ares_dns_class_fromstr(3)* function outputs the DNS class in numeric form from its string representation in *str*. The result is stored into the variable pointed to by *qclass*.

The *ares_dns_rec_type_fromstr(3)* function outputs the DNS record type in numeric form from its string representation in *str*. The result is stored into the variable pointed to by *qtype*.

The *ares_dns_rr_get_keys(3)* function retrieves a list of parameters that may be set or retrieved for the provided *type*. The count of returned keys is stored into the variable pointed to by *cnt*.

The *ares_dns_rr_key_datatype(3)* function retrieves the associated datatype for an RR parameter specified by *key*.

The *ares_dns_rr_key_to_rec_type(3)* function dereferences the provided RR parameter specified by *key* to the DNS Record Type it belongs.

The *ares_dns_opt_get_datatype(3)* function is used in association with *ares_dns_rr_set_opt(3)* and *ares_dns_rr_get_opt(3)* to retrieve the datatype of an option record contained within an RR as specified in *key* if it is known. The raw option record identifier is provided by *opt*.

The *ares_dns_opt_get_name(3)* function is used in association with *ares_dns_rr_set_opt(3)* and *ares_dns_rr_get_opt(3)* to retrieve human readable parameter name of an option record contained within an RR as specified in *key* if it is known. The raw option record identifier is provided by *opt*.

RETURN VALUES

ares_dns_rec_type_tostr(3), *ares_dns_class_tostr(3)*, *ares_dns_opcode_tostr(3)*, *ares_dns_rr_key_tostr(3)*, *ares_dns_section_tostr(3)*, *ares_dns_rcode_tostr(3)*, and *ares_dns_opt_get_name(3)* all return a human printable ASCII string, or NULL on error.

ares_dns_class_fromstr(3) and *ares_dns_rec_type_fromstr(3)* return **ARES_TRUE** on successful conversion, otherwise **ARES_FALSE**.

ares_dns_rr_get_keys(3) returns an array of keys or NULL on failure.

ares_dns_rr_key_datatype(3), *ares_dns_rr_key_to_rec_type(3)*, and *ares_dns_opt_get_datatype(3)* return their respective integer values, or 0 on failure.

AVAILABILITY

These functions were first introduced in c-ares version 1.22.0.

SEE ALSO

ares_dns_record(3), **ares_dns_rr(3)**, **ares_init(3)**