

**NAME**

`ares_parse_aaaa_reply` – Parse a reply to a DNS query of type AAAA

**SYNOPSIS**

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

```
int ares_parse_aaaa_reply(const unsigned char *abuf, int alen,  
                          struct hostent **host,  
                          struct ares_addr6ttl *addr6ttls, int *naddr6ttls);
```

**DESCRIPTION**

The `ares_parse_aaaa_reply` function parses the response to a query of type AAAA into a **struct hostent** and/or an array of **struct ares\_addr6ttl**. The parameters *abuf* and *alen* give the contents of the response. The result is stored in allocated memory and a pointer to it stored into the variable pointed to by *host*, if *host* is nonnull. It is the caller's responsibility to free the resulting host structure using `ares_free_hostent(3)` when it is no longer needed.

If *addr6ttls* and *naddr6ttls* are both nonnull, then up to *\*naddr6ttls* **struct ares\_addr6ttl** records are stored in the array pointed to by *addr6ttls*, and then *\*naddr6ttls* is set to the number of records so stored. Note that the memory for these records is supplied by the caller.

**RETURN VALUES**

`ares_parse_aaaa_reply` can return any of the following values:

**ARES\_SUCCESS**

The response was successfully parsed.

**ARES\_EBADRESP**

The response was malformed.

**ARES\_ENODATA**

The response did not contain an answer to the query.

**ARES\_ENOMEM**

Memory was exhausted.

**SEE ALSO**

`ares_gethostbyname(3)`, `ares_free_hostent(3)`

**AUTHOR**

Dominick Meglio  
Copyright 2005 by Dominick Meglio.