

**NAME**

`ares_save_options` – Save configuration values obtained from initialized `ares_channel`

**SYNOPSIS**

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

```
int ares_save_options(ares_channel channel, struct ares_options *options, int *optmask)
```

**DESCRIPTION**

The `ares_save_options(3)` function saves the channel data identified by `channel`, into the options struct identified by `options`, and saves the mask of options which are set to the integer pointer (passed by reference) identified by `optmask`.

The resultant options and `optmask` are then able to be passed directly to `ares_init_options`. When the options are no longer needed, `ares_destroy_options` should be called to free any associated memory.

**RETURN VALUES**

`ares_save_options(3)` can return any of the following values:

**ARES\_SUCCESS**

The channel data was successfully stored

**ARES\_ENOMEM**

The memory was exhausted

**ARES\_ENODATA**

The channel data identified by `channel` were invalid.

**NOTE**

Since c-ares 1.6.0 the `ares_options` struct has been "locked" meaning that it won't be extended to cover new functions. This function will remain functioning, but it can only return config data that can be represented in this config struct, which may no longer be the complete set of config options. `ares_dup(3)` will not have that restriction.

The `ares_options` struct can not handle potential IPv6 name servers the `ares_channel` might be configured to use. The `ares_save_options(3)` function will only return IPv4 servers, if any. In order to retrieve all name servers an `ares_channel` might be using, the `ares_get_servers(3)` function must be used instead.

**SEE ALSO**

`ares_destroy_options(3)`, `ares_init_options(3)`, `ares_get_servers(3)`, `ares_dup(3)`

**AVAILABILITY**

`ares_save_options(3)` was added in c-ares 1.4.0

**AUTHOR**

Brad House

Copyright 1998 by the Massachusetts Institute of Technology.