

**NAME**

`ares_reinit` – ReInitialize a resolver channel from system configuration.

**SYNOPSIS**

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

```
int ares_reinit(ares_channel_t *channel)
```

**DESCRIPTION**

The **`ares_reinit(3)`** function re-reads the system configuration and safely applies the configuration to the existing channel. System configuration will never override user-provided settings such as provided via **`ares_init_options(3)`** or **`ares_set_servers(3)`**.

Any existing queries will be automatically queued if the server they are currently assigned to is removed from the system configuration.

This function may cause additional file descriptors to be created, and existing ones to be destroyed if server configuration has changed.

**0d safety, will spawn a background thread to read the configuration and apply it. It is crucial that developers use the `ARES_OPT_SOCK_STATE_CB` or `ARES_OPT_EVENT_THREAD` so that notifications of changes are alerted. If using `ares_getsock(3)` or `ares_fds(3)`, no notification is possible which could cause a stall.**

**RETURN VALUES**

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

**ARES\_SUCCESS**

Initialization succeeded.

**ARES\_EFILE** A configuration file could not be read.

**ARES\_ENOMEM**

The process's available memory was exhausted.

**AVAILABILITY**

This function was first introduced in c-ares version 1.22.0.

**SEE ALSO**

**`ares_init(3)`, `ares_init_options(3)`, `ares_destroy(3)`, `ares_dup(3)`, `ares_library_init(3)`, `ares_set_servers(3)`, `ares_threadafety(3)`**