

nghttp21.51.0 DEV

Search docs

nghttp2 - HTTP/2 C Library

Contribution Guidelines

Security Process

Building Android binary

Tutorial: HTTP/2 client

Tutorial: HTTP/2 server

Tutorial: HPACK API

nghttp(1)

nghttpd(1)

nghttpx(1)

h2load(1)

nghttpx - HTTP/2 proxy - HOW-TO

h2load - HTTP/2 benchmarking tool - HOW-TO

Programmers' Guide

API Reference

Macros

Enums

Types (structs, unions and typedefs)

nghttp2_check_authority

nghttp2_check_header_name

nghttp2_check_header_value

nghttp2_check_header_value_rfc9113

nghttp2_check_method

nghttp2_check_path

nghttp2_hd_deflate_bound

nghttp2_hd_deflate_change_table_size

nghttp2_hd_deflate_del

nghttp2_hd_deflate_get_dynamic_table_size

nghttp2_hd_deflate_get_max_dynamic_table_size

nghttp2_hd_deflate_get_num_table_entries

nghttp2_hd_deflate_get_table_entries

nghttp2_hd_deflate_get_table_entry

nghttp2_hd_deflate_hd

nghttp2_hd_deflate_hd_vec

nghttp2_hd_deflate_new

nghttp2_hd_deflate_new2

nghttp2_hd_inflate_change_table_size

nghttp2_hd_inflate_del

nghttp2_hd_inflate_end_headers

nghttp2_hd_inflate_get_dynamic_table_size

nghttp2_hd_inflate_get_max_dynamic_table_size

nghttp2_hd_inflate_get_num_table_entries

nghttp2_hd_inflate_get_table_entry

nghttp2_select_next_protocol

Synopsis

#include <nghttp2/nghttp2.h>

```
int nghttp2_select_next_protocol(unsigned char **out, unsigned char *outlen, const unsigned char *in, unsigned int inlen)
```

A helper function for dealing with NPN in client side or ALPN in server side. The in contains peer's protocol list in preferable order. The format of in is length-prefixed and not null-terminated. For example, `{2}` and `{http/1.1}` stored in like this:

```
in[0] = 2
in[1..2] = "h2"
in[3] = 8
in[4..11] = "http/1.1"
inlen = 12
```

The selection algorithm is as follows:

1. If peer's list contains HTTP/2 protocol the library supports, it is selected and returns 1. The following step is not taken.
2. If peer's list contains `{http/1.1}`, this function selects `{http/1.1}` and returns 0. The following step is not taken.
3. This function selects nothing and returns -1 (So called non-overlap case). In this case, out and outlen are left untouched.

Selecting `{2}` means that `{h2}` is written into *out and its length (which is 2) is assigned to *outlen.

For ALPN, refer to <https://tools.ietf.org/html/rfc7301>

See <http://technotes.googlecode.com/git/nextprotoneg.html> for more details about NPN.

For NPN, to use this method you should do something like:

```
static int select_next_proto_cb(SSL* ssl,
                               unsigned char **out,
                               unsigned char *outlen,
                               const unsigned char *in,
                               unsigned int inlen,
                               void *arg)
{
    int rv;
    rv = nghttp2_select_next_protocol(out, outlen, in, inlen);
    if (rv == -1) {
        return SSL_TLSEXT_ERR_NOACK;
    }
    if (rv == 1) {
        ((MyType*)arg)->http2_selected = 1;
    }
    return SSL_TLSEXT_ERR_OK;
}
...
SSL_CTX_set_next_proto_select_cb(ssl_ctx, select_next_proto_cb, my_obj);
```

Previous

Next