

Howto write an HTTP client

Table of contents

- [Client initialization](#)
- [Connection initialization](#)
 - [SSL related functions](#)
- [Setting HTTP headers \(optional\)](#)
 - [Setting an header with uniq key](#)
 - [Setting headers with the same key](#)
 - [HTTP authorization](#)
- [Request the network resource](#)
 - [HTTP GET method](#)
 - [HTTP POST method](#)
 - [MIME attachments](#)
- [Fetch and print out the result](#)
- [Connection cleanup](#)
- [Client cleanup](#)

Client initialization

```
int main(int argc, char **argv)
{
    herrord_t status;
    size_t len;
    httpc_conn_t conn;

    if (argc < 2)
    {
        fprintf(stderr, "usage: %s <url> [nanoHTTP params]\n", argv[0]);
        exit(1);
    }

    if ((status = httpc_init(argc, argv)) != H_OK)
    {
        fprintf(stderr, "Cannot init nanoHTTP client (%s)\n", httpc_message(status));
        httpc_release(status);
        exit(1);
    }
}
```

Connection initialization

```
if (!(conn = httpc_new()))
{
    fprintf(stderr, "Cannot create nanoHTTP client connection\n");
    httpc_destroy();
    exit(1);
}
```

SSL related functions

T.B.D.

How to use SSL with nanoHTTP

Setting HTTP headers (optional)

Setting an header with uniq key

```
httpc_set_header(conn, "my-key", "my-value");
```

Setting headers with the same key

```
httpc_add_header(conn, "Cookie", "name1:value1");  
httpc_add_header(conn, "Cookie", "name2:value2");
```

Please see [General Header Fields](#) and [Request Header Fields](#) for more information.

HTTP authorization

```
httpc_set_basic_authorization(conn, "username", "password");
```

```
httpc_set_basic_proxy_authorization(conn, "username", "password");
```

Request the network resource

HTTP GET method

```
if ((status = httpc_get(conn, &result, argv[1])) != H_OK)  
{  
    fprintf(stderr, "nanoHTTP client connection failed (%s)\n", httpc_message(status));  
    httpc_release(status);  
    httpc_destroy();  
    exit(1);  
}
```

HTTP POST method

```
if ((status = httpc_post_begin(conn, argv[1])) != H_OK)  
{  
    fprintf(stderr, "nanoHTTP client connection failed (%s)\n", httpc_message(status));  
    httpc_release(status);  
    httpc_destroy();  
    exit(1);  
}  
  
if ((status = http_output_stream_write(conn->out, buffer, len)) != H_OK)  
{  
    fprintf(stderr, "nanoHTTP client sending POST data failed (%s)\n", httpc_message(status));  
    httpc_release(status);  
    httpc_destroy();  
    exit(1);  
}  
  
if ((status = httpc_post_end(conn, &result)) != H_OK)  
{  
    fprintf(stderr, "nanoHTTP client POST failed (%s)\n", httpc_message(status));  
    httpc_release(status);  
    httpc_destroy();  
    exit(1);  
}
```

MIME attachments

T.B.D.

nanoHTTP MIME attachments

Fetch and print out the result

```
while (http_input_stream_is_ready(res->in))  
{
```


```
len = http_input_stream_read(res->in, buffer, MAX_BUFFER_SIZE);  
fwrite(buffer, len, 1, stdout);  
}
```

Connection cleanup

```
hresponse_free(res);
```

Client cleanup

```
httpc_free(conn);  
exit(0);  
}
```

Generated on Thu Jan 25 23:36:01 2007 for cssoap by  1.4.6