NAME

OSSL_HTTP_adapt_proxy, OSSL_parse_url, OSSL_HTTP_parse_url, OCSP_parse_url - http utility functions

SYNOPSIS

```
#include <openssl/http.h>

const char *OSSL_HTTP_adapt_proxy(const char *proxy, const char *no_proxy,
                                  const char *server, int use_ssl);

int OSSL_parse_url(const char *url, char **pscheme, char **puser, char **phost,
                   char **pport, int *pport_num,
                   char **ppath, char **pquery, char **pfrag);
int OSSL_HTTP_parse_url(const char *url,
                        int *pssl, char **puser, char **phost,
                        char **pport, int *pport_num,
                        char **ppath, char **pquery, char **pfrag);
```

The following functions have been deprecated since OpenSSL 3.0, and can be hidden entirely by defining **OPENSSL_API_COMPAT** with a suitable version value, see **openssl_user_macros**(7):

```
int OCSP_parse_url(char *url, char **phost, char **pport, char **ppath,
                   int *pssl);
```

DESCRIPTION

OSSL_HTTP_adapt_proxy() takes an optional proxy hostname *proxy* and returns it transformed according to the optional *no_proxy* parameter, *server*, *use_ssl*, and the applicable environment variable, as follows. If *proxy* is NULL, take any default value from the "http_proxy" environment variable, or from "https_proxy" if *use_ssl* is nonzero. If this still does not yield a proxy hostname, take any further default value from the "HTTP_PROXY" environment variable, or from "HTTPS_PROXY" if *use_ssl* is nonzero. If *no_proxy* is NULL, take any default exclusion value from the "no_proxy" environment variable, or else from "NO_PROXY". Return the determined proxy hostname unless the exclusion contains *server*. Otherwise return NULL.

OSSL_parse_url() parses its input string *url* as a URL of the form "[scheme://][userinfo@]host[:port][/path][?query][#fragment]" and splits it up into scheme, userinfo, host, port, path, query, and fragment components. The host (or server) component may be a DNS name or an IP address where IPv6 addresses should be enclosed in square brackets "[" and "]". The port component is optional and defaults to 0. If given, it must be in decimal form. If the *pport_num* argument is not NULL the integer value of the port number is assigned to **pport_num* on success. The

path component is also optional and defaults to "/". Each non-NULL result pointer argument *pscheme*, *puser*, *phost*, *pport*, *ppath*, *pquery*, and *pfrag*, is assigned the respective url component. On success, they are guaranteed to contain non-NULL string pointers, else NULL. It is the responsibility of the caller to free them using **OPENSSL_free**(3). If *pquery* is NULL, any given query component is handled as part of the path. A string returned via **ppath* is guaranteed to begin with a "/" character. For absent scheme, userinfo, port, query, and fragment components an empty string is provided.

OSSL_HTTP_parse_url() is a special form of **OSSL_parse_url()** where the scheme, if given, must be "http" or "https". If *pssl* is not NULL, **pssl* is assigned 1 in case parsing was successful and the scheme is "https", else 0. The port component is optional and defaults to 443 if the scheme is "https", else 80. Note that relative paths must be given with a leading "/", otherwise the first path element is interpreted as the hostname.

Calling the deprecated function OCSP_parse_url(url, host, port, path, ssl) is equivalent to OSSL_HTTP_parse_url(url, ssl, NULL, host, port, NULL, path, NULL, NULL).

RETURN VALUES

OSSL_HTTP_adapt_proxy() returns NULL if no proxy is to be used, otherwise a constant proxy hostname string, which is either the proxy name handed in or an environment variable value.

OSSL_parse_url(), **OSSL_HTTP_parse_url()**, and **OCSP_parse_url()** return 1 on success, 0 on error.

SEE ALSO

OSSL_HTTP_transfer(3)

HISTORY

OSSL_HTTP_adapt_proxy(), **OSSL_parse_url()** and **OSSL_HTTP_parse_url()** were added in OpenSSL 3.0. **OCSP_parse_url()** was deprecated in OpenSSL 3.0.

COPYRIGHT