ssl

SSL context options

SSL context option listing

ENTITY

Context options for `ssl://` and `tls://` transports.

ENTITY

`peer_name` `string`
Peer name to be used. If this value is not set, then the name is guessed based on the hostname used when opening the stream.

`verify_peer` `bool`
Require verification of SSL certificate used.

Defaults to ENTITY.

`verify_peer_name` `bool`
Require verification of peer name.

Defaults to ENTITY.

`allow_self_signed` `bool`
Allow self-signed certificates. Requires `verify_peer`.

Defaults to ENTITY

`cafile` `string`
Location of Certificate Authority file on local filesystem which should be used with the `verify_peer` context option to authenticate the identity of the remote peer.

`capath` `string`
If `cafile` is not specified or if the certificate is not found there, the directory pointed to by `capath` is searched for a suitable certificate. `capath` must be a correctly hashed certificate directory.

`local_cert` `string`
Path to local certificate file on filesystem. It must be a PEM encoded file which contains your certificate and private key. It can optionally contain the certificate chain of issuers. The private key also may be contained in a separate file specified by `local_pk`.

`local_pk` `string`
Path to local private key file on filesystem in case of separate files for certificate (`local_cert`) and private key.

`passphrase` `string`
Passphrase with which your `local_cert` file was encoded.

`verify_depth` `int`
Abort if the certificate chain is too deep.

Defaults to no verification.

`ciphers` `string`
Sets the list of available ciphers. The format of the string is described in ciphers(1).

Defaults to `DEFAULT`.

`capture_peer_cert` `bool`
If set to ENTITY a `peer_certificate` context option will be created containing the peer certificate.

`capture_peer_cert_chain` `bool`
If set to ENTITY a `peer_certificate_chain` context option will be created containing the certificate chain.

`SNI_enabled` `bool`
If set to ENTITY server name indication will be enabled. Enabling SNI allows multiple certificates on the same IP address.

`SNI_server_certs` `array`
An array of server names and their corresponding certificates to be used for SNI. The keys are the server names and the values are the paths to the certificate files on the local filesystem. The certificate files must be PEM encoded and contain both the certificate and private key.

Available as of PHP 5.6.0.

`alpn_protocols` `array`
An array of application layer protocol names to be used for ALPN (Application-Layer Protocol Negotiation). The values are the protocol names as strings (e.g. "http/1.1", "h2").

Available as of PHP 7.0.0 and OpenSSL 1.0.2.

`no_ticket` `bool`
If set, disable TLS session tickets. This can help to enhance security by providing Perfect Forward Secrecy (PFS).

Available as of PHP 5.3.6.

`disable_compression` `bool`
If set, disable TLS compression. This can help mitigate the CRIME attack vector.

`peer_fingerprint` `string` | `array`
Aborts when the remote certificate digest doesn't match the specified hash.

When a `string` is used, the length will determine which hashing algorithm is applied, either "md5" (32) or "sha1" (40).

When an `array` is used, the keys indicate the hashing algorithm name and each corresponding value is the expected digest.

`security_level` `int`
Sets the security level. If not specified the library default security level is used. The security levels are described in SSL_CTX_get_security_level(3).

Available as of PHP 7.2.0 and OpenSSL 1.1.0.

`min_proto_version` `int`
Sets the minimum protocol version allowed. If not specified the library default minimum protocol version is used. The protocol versions are described in SSL_CTX_set_min_proto_version(3).

Available as of PHP 7.3.0 and OpenSSL 1.1.0.

`max_proto_version` `int`
Sets the maximum protocol version allowed. If not specified the library default maximum protocol version is used. The protocol versions are described in SSL_CTX_set_max_proto_version(3).

Available as of PHP 7.3.0 and OpenSSL 1.1.0.

ENTITY

ENTITY	ENTITY
7.3.0	Added `min_proto_version` and `max_proto_version`. Requires OpenSSL ENTITY= 1.1.0.
7.2.0	Added `security_level`. Requires OpenSSL ENTITY= 1.1.0.
7.0.0	Added `alpn_protocols`. Requires OpenSSL ENTITY= 1.0.2.

ENTITY	ENTITY
5.6.0	Added `SNI_server_certs`.
5.3.6	Added `no_ticket`.

ENTITY

> Because `ssl://` is the underlying transport for the `https://` and `ftps://` wrappers, any context options which apply to `ssl://` also apply to `https://` and `ftps://`.

> For SNI (Server Name Indication) to be available, then PHP must be compiled with OpenSSL 0.9.8j or greater. Use the `OPENSSL_TLSEXT_SERVER_NAME` to determine whether SNI is supported.

ENTITY

□□□ #2
□ □□□ □□□ 2026-10-31 17:38:20 UTC
□ □□□ □□□ 2026-10-31 17:38:22 UTC