**Transport Layer Security**

From Wikipedia, the free encyclopedia

[Jump to navigation](https://en.wikipedia.org/wiki/Transport_Layer_Security#mw-head) [Jump to search](https://en.wikipedia.org/wiki/Transport_Layer_Security#searchInput)

|  |
| --- |
| [**Internet protocol suite**](https://en.wikipedia.org/wiki/Internet_protocol_suite) |
| [**Application layer**](https://en.wikipedia.org/wiki/Application_layer) |
| * [BGP](https://en.wikipedia.org/wiki/Border_Gateway_Protocol) * [DHCP](https://en.wikipedia.org/wiki/Dynamic_Host_Configuration_Protocol) * [DNS](https://en.wikipedia.org/wiki/Domain_Name_System) * [FTP](https://en.wikipedia.org/wiki/File_Transfer_Protocol) * [HTTP](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol) * [HTTPS](https://en.wikipedia.org/wiki/HTTPS) * [IMAP](https://en.wikipedia.org/wiki/Internet_Message_Access_Protocol) * [LDAP](https://en.wikipedia.org/wiki/Lightweight_Directory_Access_Protocol) * [MGCP](https://en.wikipedia.org/wiki/Media_Gateway_Control_Protocol) * [MQTT](https://en.wikipedia.org/wiki/MQTT) * [NNTP](https://en.wikipedia.org/wiki/Network_News_Transfer_Protocol) * [NTP](https://en.wikipedia.org/wiki/Network_Time_Protocol) * [POP](https://en.wikipedia.org/wiki/Post_Office_Protocol) * [PTP](https://en.wikipedia.org/wiki/Precision_Time_Protocol) * [ONC/RPC](https://en.wikipedia.org/wiki/Open_Network_Computing_Remote_Procedure_Call) * [RTP](https://en.wikipedia.org/wiki/Real-time_Transport_Protocol) * [RTSP](https://en.wikipedia.org/wiki/Real_Time_Streaming_Protocol) * [RIP](https://en.wikipedia.org/wiki/Routing_Information_Protocol) * [SIP](https://en.wikipedia.org/wiki/Session_Initiation_Protocol) * [SMTP](https://en.wikipedia.org/wiki/Simple_Mail_Transfer_Protocol) * [SNMP](https://en.wikipedia.org/wiki/Simple_Network_Management_Protocol) * [SSH](https://en.wikipedia.org/wiki/Secure_Shell) * [Telnet](https://en.wikipedia.org/wiki/Telnet) * TLS/SSL * [XMPP](https://en.wikipedia.org/wiki/XMPP) * [*more...*](https://en.wikipedia.org/wiki/Category:Application_layer_protocols) |
| [**Transport layer**](https://en.wikipedia.org/wiki/Transport_layer) |
| * [TCP](https://en.wikipedia.org/wiki/Transmission_Control_Protocol) * [UDP](https://en.wikipedia.org/wiki/User_Datagram_Protocol) * [DCCP](https://en.wikipedia.org/wiki/Datagram_Congestion_Control_Protocol) * [SCTP](https://en.wikipedia.org/wiki/Stream_Control_Transmission_Protocol) * [RSVP](https://en.wikipedia.org/wiki/Resource_Reservation_Protocol) * [*more...*](https://en.wikipedia.org/wiki/Category:Transport_layer_protocols) |
| [**Internet layer**](https://en.wikipedia.org/wiki/Internet_layer) |
| * [IP](https://en.wikipedia.org/wiki/Internet_Protocol)   + [IPv4](https://en.wikipedia.org/wiki/IPv4)   + [IPv6](https://en.wikipedia.org/wiki/IPv6) * [ICMP](https://en.wikipedia.org/wiki/Internet_Control_Message_Protocol) * [ICMPv6](https://en.wikipedia.org/wiki/Internet_Control_Message_Protocol_for_IPv6) * [ECN](https://en.wikipedia.org/wiki/Explicit_Congestion_Notification) * [IGMP](https://en.wikipedia.org/wiki/Internet_Group_Management_Protocol) * [IPsec](https://en.wikipedia.org/wiki/IPsec) * [*more...*](https://en.wikipedia.org/wiki/Category:Internet_layer_protocols) |
| [**Link layer**](https://en.wikipedia.org/wiki/Link_layer) |
| * [ARP](https://en.wikipedia.org/wiki/Address_Resolution_Protocol) * [NDP](https://en.wikipedia.org/wiki/Neighbor_Discovery_Protocol) * [OSPF](https://en.wikipedia.org/wiki/Open_Shortest_Path_First) * [Tunnels](https://en.wikipedia.org/wiki/Tunneling_protocol)   + [L2TP](https://en.wikipedia.org/wiki/Layer_2_Tunneling_Protocol) * [PPP](https://en.wikipedia.org/wiki/Point-to-Point_Protocol) * [MAC](https://en.wikipedia.org/wiki/Medium_access_control)   + [Ethernet](https://en.wikipedia.org/wiki/Ethernet)   + [Wi-Fi](https://en.wikipedia.org/wiki/Wi-Fi)   + [DSL](https://en.wikipedia.org/wiki/Digital_subscriber_line)   + [ISDN](https://en.wikipedia.org/wiki/Integrated_Services_Digital_Network)   + [FDDI](https://en.wikipedia.org/wiki/Fiber_Distributed_Data_Interface) * [*more...*](https://en.wikipedia.org/wiki/Category:Link_protocols) |
| * [v](https://en.wikipedia.org/wiki/Template:IPstack) * [t](https://en.wikipedia.org/wiki/Template_talk:IPstack) * [e](https://en.wikipedia.org/w/index.php?title=Template:IPstack&action=edit) |

**Transport Layer Security** (**TLS**), and its now-deprecated predecessor, **Secure Sockets Layer** (**SSL**),[[1]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-1) are [cryptographic protocols](https://en.wikipedia.org/wiki/Cryptographic_protocol) designed to provide [communications security](https://en.wikipedia.org/wiki/Communications_security) over a [computer network](https://en.wikipedia.org/wiki/Computer_network).[[2]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RFC5246-2) Several versions of the protocols find widespread use in applications such as [web browsing](https://en.wikipedia.org/wiki/Web_navigation), [email](https://en.wikipedia.org/wiki/Email), [instant messaging](https://en.wikipedia.org/wiki/Instant_messaging), and [voice over IP](https://en.wikipedia.org/wiki/Voice_over_IP) (VoIP). [Websites](https://en.wikipedia.org/wiki/Website) can use TLS to secure all communications between their [servers](https://en.wikipedia.org/wiki/Server_(computing)) and [web browsers](https://en.wikipedia.org/wiki/Web_browser).

The TLS protocol aims primarily to provide [privacy](https://en.wikipedia.org/wiki/Privacy) and [data integrity](https://en.wikipedia.org/wiki/Data_integrity) between two or more communicating computer applications.[[2]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RFC5246-2):3 When secured by TLS, connections between a client (e.g., a web browser) and a server (e.g., wikipedia.org) should have one or more of the following properties:

* The connection is *private* (or *secure*) because [symmetric cryptography](https://en.wikipedia.org/wiki/Symmetric-key_algorithm) is used to [encrypt](https://en.wikipedia.org/wiki/Encryption) the data transmitted. The [keys](https://en.wikipedia.org/wiki/Key_(cryptography)) for this symmetric encryption are generated uniquely for each connection and are based on a [shared secret](https://en.wikipedia.org/wiki/Shared_secret) that was negotiated at the start of the [session](https://en.wikipedia.org/wiki/Session_(computer_science)) (see [§ TLS handshake](https://en.wikipedia.org/wiki/Transport_Layer_Security#TLS_handshake)). The server and client negotiate the details of which encryption algorithm and cryptographic keys to use before the first [byte](https://en.wikipedia.org/wiki/Byte) of data is transmitted (see [§ Algorithm](https://en.wikipedia.org/wiki/Transport_Layer_Security#Algorithm) below). The negotiation of a shared secret is both secure (the negotiated secret is unavailable to [eavesdroppers](https://en.wikipedia.org/wiki/Eavesdropping) and cannot be obtained, even by an attacker who places themselves in the middle of the connection) and reliable (no attacker can modify the communications during the negotiation without being detected).
* The identity of the communicating parties can be *authenticated* using [public-key cryptography](https://en.wikipedia.org/wiki/Public-key_cryptography). This authentication can be made optional, but is generally required for at least one of the parties (typically the server).
* The connection is *reliable* because each message transmitted includes a message integrity check using a [message authentication code](https://en.wikipedia.org/wiki/Message_authentication_code) to prevent undetected loss or alteration of the data during [transmission](https://en.wikipedia.org/wiki/Data_transmission).[[2]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RFC5246-2):3

In addition to the properties above, careful [configuration](https://en.wikipedia.org/wiki/Computer_configuration) of TLS can provide additional privacy-related properties such as [forward secrecy](https://en.wikipedia.org/wiki/Forward_secrecy), ensuring that any future disclosure of encryption keys cannot be used to decrypt any TLS communications recorded in the past.[[3]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-3)

TLS supports many different methods for exchanging keys, encrypting data, and authenticating message integrity (see [§ Algorithm](https://en.wikipedia.org/wiki/Transport_Layer_Security#Algorithm) below). As a result, secure configuration of TLS involves many configurable parameters, and not all choices provide all of the privacy-related properties described in the list above (see the [§ Key exchange](https://en.wikipedia.org/wiki/Transport_Layer_Security#keyexchange-table) (authentication), [§ Cipher security](https://en.wikipedia.org/wiki/Transport_Layer_Security#cipher-table), and [§ Data integrity](https://en.wikipedia.org/wiki/Transport_Layer_Security#integrity-table) tables).

Attempts have been made to subvert aspects of the communications security that TLS seeks to provide, and the protocol has been revised several times to address these security threats (see [§ Security](https://en.wikipedia.org/wiki/Transport_Layer_Security#Security)). Developers of web browsers have also revised their products to defend against potential security weaknesses after these were discovered (see [TLS/SSL support history of web browsers](https://en.wikipedia.org/wiki/Transport_Layer_Security#Web_browsers)).[[4]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-:0-4)

The TLS protocol comprises two layers: the [TLS record](https://en.wikipedia.org/wiki/Transport_Layer_Security#TLS_record) and the [TLS handshake](https://en.wikipedia.org/wiki/Transport_Layer_Security#TLS_handshake) protocols.

TLS is a proposed Internet Engineering Task Force ([IETF](https://en.wikipedia.org/wiki/IETF)) [standard](https://en.wikipedia.org/wiki/Internet_Standard), first defined in 1999, and the current version is TLS 1.3 defined in [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [8446](https://tools.ietf.org/html/rfc8446) (August 2018). TLS builds on the earlier SSL specifications (1994, 1995, 1996) developed by [Netscape Communications](https://en.wikipedia.org/wiki/Netscape)[[5]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RFC6101-5) for adding the [HTTPS](https://en.wikipedia.org/wiki/HTTP_Secure) protocol to their [Navigator](https://en.wikipedia.org/wiki/Netscape_Navigator) web browser.



**Contents**

* [1 Description](https://en.wikipedia.org/wiki/Transport_Layer_Security#Description)
* [2 History and development](https://en.wikipedia.org/wiki/Transport_Layer_Security#History_and_development)
  + [2.1 Secure Data Network System](https://en.wikipedia.org/wiki/Transport_Layer_Security#Secure_Data_Network_System)
  + [2.2 Secure Network Programming](https://en.wikipedia.org/wiki/Transport_Layer_Security#Secure_Network_Programming)
  + [2.3 SSL 1.0, 2.0, and 3.0](https://en.wikipedia.org/wiki/Transport_Layer_Security#SSL_1.0,_2.0,_and_3.0)
  + [2.4 TLS 1.0](https://en.wikipedia.org/wiki/Transport_Layer_Security#TLS_1.0)
  + [2.5 TLS 1.1](https://en.wikipedia.org/wiki/Transport_Layer_Security#TLS_1.1)
  + [2.6 TLS 1.2](https://en.wikipedia.org/wiki/Transport_Layer_Security#TLS_1.2)
  + [2.7 TLS 1.3](https://en.wikipedia.org/wiki/Transport_Layer_Security#TLS_1.3)
* [3 Digital certificates](https://en.wikipedia.org/wiki/Transport_Layer_Security#Digital_certificates)
  + [3.1 Certificate authorities](https://en.wikipedia.org/wiki/Transport_Layer_Security#Certificate_authorities)
* [4 Algorithms](https://en.wikipedia.org/wiki/Transport_Layer_Security#Algorithms)
  + [4.1 Key exchange or key agreement](https://en.wikipedia.org/wiki/Transport_Layer_Security#Key_exchange_or_key_agreement)
  + [4.2 Cipher](https://en.wikipedia.org/wiki/Transport_Layer_Security#Cipher)
  + [4.3 Data integrity](https://en.wikipedia.org/wiki/Transport_Layer_Security#Data_integrity)
* [5 Applications and adoption](https://en.wikipedia.org/wiki/Transport_Layer_Security#Applications_and_adoption)
  + [5.1 Websites](https://en.wikipedia.org/wiki/Transport_Layer_Security#Websites)
  + [5.2 Web browsers](https://en.wikipedia.org/wiki/Transport_Layer_Security#Web_browsers)
  + [5.3 Libraries](https://en.wikipedia.org/wiki/Transport_Layer_Security#Libraries)
  + [5.4 Other uses](https://en.wikipedia.org/wiki/Transport_Layer_Security#Other_uses)
* [6 Security](https://en.wikipedia.org/wiki/Transport_Layer_Security#Security)
  + [6.1 SSL 2.0](https://en.wikipedia.org/wiki/Transport_Layer_Security#SSL_2.0)
  + [6.2 SSL 3.0](https://en.wikipedia.org/wiki/Transport_Layer_Security#SSL_3.0)
  + [6.3 TLS](https://en.wikipedia.org/wiki/Transport_Layer_Security#TLS)
  + [6.4 Attacks against TLS/SSL](https://en.wikipedia.org/wiki/Transport_Layer_Security#Attacks_against_TLS/SSL)
  + [6.5 Forward secrecy](https://en.wikipedia.org/wiki/Transport_Layer_Security#Forward_secrecy)
  + [6.6 TLS interception](https://en.wikipedia.org/wiki/Transport_Layer_Security#TLS_interception)
* [7 Protocol details](https://en.wikipedia.org/wiki/Transport_Layer_Security#Protocol_details)
  + [7.1 TLS handshake](https://en.wikipedia.org/wiki/Transport_Layer_Security#TLS_handshake)
  + [7.2 TLS record](https://en.wikipedia.org/wiki/Transport_Layer_Security#TLS_record)
* [8 Support for name-based virtual servers](https://en.wikipedia.org/wiki/Transport_Layer_Security#Support_for_name-based_virtual_servers)
* [9 Standards](https://en.wikipedia.org/wiki/Transport_Layer_Security#Standards)
  + [9.1 Primary standards](https://en.wikipedia.org/wiki/Transport_Layer_Security#Primary_standards)
  + [9.2 Extensions](https://en.wikipedia.org/wiki/Transport_Layer_Security#Extensions)
  + [9.3 Informational RFCs](https://en.wikipedia.org/wiki/Transport_Layer_Security#Informational_RFCs)
* [10 See also](https://en.wikipedia.org/wiki/Transport_Layer_Security#See_also)
* [11 References](https://en.wikipedia.org/wiki/Transport_Layer_Security#References)
* [12 Further reading](https://en.wikipedia.org/wiki/Transport_Layer_Security#Further_reading)
* [13 External links](https://en.wikipedia.org/wiki/Transport_Layer_Security#External_links)

**Description**

[Client-server](https://en.wikipedia.org/wiki/Client%E2%80%93server_model) applications use the TLS [protocol](https://en.wikipedia.org/wiki/Proprietary_protocol) to communicate across a network in a way designed to prevent eavesdropping and [tampering](https://en.wikipedia.org/wiki/Tamper-evident).

Since applications can communicate either with or without TLS (or SSL), it is necessary for the [client](https://en.wikipedia.org/wiki/Client_(computing)) to indicate to the [server](https://en.wikipedia.org/wiki/Server_(computing)) the setup of a TLS connection.[[6]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-6) One of the main ways of achieving this is to use a different [port number](https://en.wikipedia.org/wiki/Port_(computer_networking)) for TLS connections, for example port 443 for [HTTPS](https://en.wikipedia.org/wiki/HTTPS). Another mechanism is for the client to make a protocol-specific request to the server to switch the connection to TLS; for example, by making a [STARTTLS](https://en.wikipedia.org/wiki/Opportunistic_TLS) request when using the mail and [news](https://en.wikipedia.org/wiki/Usenet) protocols.

Once the client and server have agreed to use TLS, they negotiate a [stateful](https://en.wikipedia.org/wiki/State_(computer_science)) connection by using a [handshaking](https://en.wikipedia.org/wiki/Transport_Layer_Security#Protocol_details) procedure.[[7]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-7) The protocols use a handshake with an [asymmetric cipher](https://en.wikipedia.org/wiki/Asymmetric_cipher) to establish not only cipher settings but also a session-specific shared key with which further communication is encrypted using a [symmetric cipher](https://en.wikipedia.org/wiki/Symmetric_cipher). During this handshake, the client and server agree on various parameters used to establish the connection's security:

* The handshake begins when a client connects to a TLS-enabled server requesting a secure connection and the client presents a list of supported [cipher suites](https://en.wikipedia.org/wiki/Cipher_suite) ([ciphers](https://en.wikipedia.org/wiki/Encryption) and [hash functions](https://en.wikipedia.org/wiki/Cryptographic_hash_function)).
* From this list, the server picks a cipher and hash function that it also supports and notifies the client of the decision.
* The server usually then provides identification in the form of a [digital certificate](https://en.wikipedia.org/wiki/Public_key_certificate). The certificate contains the [server name](https://en.wikipedia.org/wiki/Hostname), the trusted [certificate authority](https://en.wikipedia.org/wiki/Certificate_authority) (CA) that vouches for the authenticity of the certificate, and the server's public encryption key.
* The client confirms the validity of the certificate before proceeding.
* To generate the session keys used for the secure connection, the client either:
  + encrypts a [random number](https://en.wikipedia.org/wiki/Random_number_generation) with the server's public key and sends the result to the server (which only the server should be able to decrypt with its private key); both parties then use the random number to generate a unique session key for subsequent encryption and decryption of data during the session
  + uses [Diffie–Hellman key exchange](https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange) to securely generate a random and unique session key for encryption and decryption that has the additional property of forward secrecy: if the server's private key is disclosed in future, it cannot be used to decrypt the current session, even if the session is intercepted and recorded by a third party.

This concludes the handshake and begins the secured connection, which is encrypted and decrypted with the session key until the connection closes. If any one of the above steps fails, then the TLS handshake fails and the connection is not created.

TLS and SSL do not fit neatly into any single layer of the [OSI model](https://en.wikipedia.org/wiki/OSI_model) or the [TCP/IP model](https://en.wikipedia.org/wiki/Internet_protocol_suite).[[8]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ccnp-8)[[9]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-stackex_layer-9) TLS runs "on top of some reliable transport protocol (e.g., TCP),"[[10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-rfc_intro-10) which would imply that it is above the [transport layer](https://en.wikipedia.org/wiki/Transport_layer). It serves encryption to higher layers, which is normally the function of the [presentation layer](https://en.wikipedia.org/wiki/Presentation_layer). However, applications generally use TLS as if it were a transport layer,[[8]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ccnp-8)[[9]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-stackex_layer-9) even though applications using TLS must actively control initiating TLS handshakes and handling of exchanged authentication certificates.[[10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-rfc_intro-10)

**History and development**

| SSL and TLS protocols | | |
| --- | --- | --- |
| **Protocol** | **Published** | **Status** |
| **SSL 1.0** | Unpublished | Unpublished |
| **SSL 2.0** | 1995 | Deprecated in 2011 ([RFC 6176](https://tools.ietf.org/html/rfc6176)) |
| **SSL 3.0** | 1996 | Deprecated in 2015 ([RFC 7568](https://tools.ietf.org/html/rfc7568)) |
| **TLS 1.0** | 1999 | Deprecated in 2020[[11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-:2-11)[[12]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-:3-12)[[13]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-:4-13) |
| **TLS 1.1** | 2006 | Deprecated in 2020[[11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-:2-11)[[12]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-:3-12)[[13]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-:4-13) |
| **TLS 1.2** | 2008 |  |
| **TLS 1.3** | 2018 |  |

**Secure Data Network System**

The Transport Layer Security Protocol (TLS), together with several other basic network security platforms, was developed through a joint initiative begun in August 1986, among the National Security Agency, the National Bureau of Standards, the Defense Communications Agency, and twelve communications and computer corporations who initiated a special project called the Secure Data Network System (SDNS).[[14]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-14) The program was described in September 1987 at the 10th National Computer Security Conference in an extensive set of published papers. The innovative research program focused on designing the next generation of secure computer communications network and product specifications to be implemented for applications on public and private internets. It was intended to complement the rapidly emerging new OSI internet standards moving forward both in the U.S. government's GOSIP Profiles and in the huge ITU-ISO JTC1 internet effort internationally. Originally known as the SP4 protocol, it was renamed TLS and subsequently published in 1995 as international standard ITU-T X.274| ISO/IEC 10736:1995.

**Secure Network Programming**

Early research efforts towards transport layer security included the [Secure Network Programming](https://en.wikipedia.org/wiki/Secure_Network_Programming) (SNP) [application programming interface](https://en.wikipedia.org/wiki/Application_programming_interface) (API), which in 1993 explored the approach of having a secure transport layer API closely resembling [Berkeley sockets](https://en.wikipedia.org/wiki/Berkeley_sockets), to facilitate retrofitting pre-existing network applications with security measures.[[15]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Woo94-15)

**SSL 1.0, 2.0, and 3.0**

Netscape developed the original SSL protocols, and [Taher Elgamal](https://en.wikipedia.org/wiki/Taher_Elgamal), chief scientist at Netscape Communications from 1995 to 1998, has been described as the "father of SSL".[[16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Messmer-16)[[17]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Greene-17)[[18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Oppliger-18)[[19]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-19) SSLVersion 1.0 was never publicly released because of serious security flaws in the protocol. Version 2.0, released in February 1995, contained a number of security flaws which necessitated the design of version 3.0.[[20]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-20)[[18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Oppliger-18) Released in 1996, SSL version 3.0 represented a complete redesign of the protocol produced by [Paul Kocher](https://en.wikipedia.org/wiki/Paul_Kocher) working with Netscape engineers Phil Karlton and Alan Freier, with a reference implementation by Christopher Allen and Tim Dierks of Consensus Development. Newer versions of SSL/TLS are based on SSL 3.0. The 1996 draft of SSL 3.0 was published by IETF as a historical document in [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6101](https://tools.ietf.org/html/rfc6101).

SSL 2.0 was deprecated in 2011 by [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6176](https://tools.ietf.org/html/rfc6176). In 2014, SSL 3.0 was found to be vulnerable to the [POODLE](https://en.wikipedia.org/wiki/POODLE) attack that affects all [block ciphers](https://en.wikipedia.org/wiki/Block_cipher) in SSL; [RC4](https://en.wikipedia.org/wiki/RC4), the only non-block cipher supported by SSL 3.0, is also feasibly broken as used in SSL 3.0.[[21]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Poodle-21) SSL 3.0 was deprecated in June 2015 by [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7568](https://tools.ietf.org/html/rfc7568).

**TLS 1.0**

TLS 1.0 was first defined in [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [2246](https://tools.ietf.org/html/rfc2246) in January 1999 as an upgrade of SSL Version 3.0, and written by Christopher Allen and Tim Dierks of Consensus Development. As stated in the RFC, "the differences between this protocol and SSL 3.0 are not dramatic, but they are significant enough to preclude interoperability between TLS 1.0 and SSL 3.0". Tim Dierks later wrote that these changes, and the renaming from "SSL" to "TLS", were a face-saving gesture to Microsoft, "so it wouldn't look [like] the IETF was just rubberstamping Netscape's protocol".[[22]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-22)

TLS 1.0 includes a means by which a TLS implementation can downgrade the connection to SSL 3.0, thus weakening security.[[23]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-urlnvlpubs.nist.gov-23):1–2

The [PCI Council](https://en.wikipedia.org/wiki/Payment_Card_Industry_Security_Standards_Council) suggested that organizations migrate from TLS 1.0 to TLS 1.1 or higher before June 30, 2018.[[24]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-24)[[25]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-25) In October 2018, [Apple](https://en.wikipedia.org/wiki/Apple_Inc.), [Google](https://en.wikipedia.org/wiki/Google), [Microsoft](https://en.wikipedia.org/wiki/Microsoft), and [Mozilla](https://en.wikipedia.org/wiki/Mozilla) jointly announced they would deprecate TLS 1.0 and 1.1 in March 2020.[[11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-:2-11)

**TLS 1.1**

TLS 1.1 was defined in [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4346](https://tools.ietf.org/html/rfc4346) in April 2006.[[26]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-26) It is an update from TLS version 1.0. Significant differences in this version include:

* Added protection against [cipher-block chaining](https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation#Cipher_Block_Chaining_(CBC)) (CBC) attacks.
  + The implicit [initialization vector](https://en.wikipedia.org/wiki/Initialization_vector) (IV) was replaced with an explicit IV.
  + Change in handling of [padding errors](https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation#Padding).
* Support for [IANA](https://en.wikipedia.org/wiki/Internet_Assigned_Numbers_Authority) registration of parameters.[[23]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-urlnvlpubs.nist.gov-23):2

**TLS 1.2**

TLS 1.2 was defined in [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5246](https://tools.ietf.org/html/rfc5246) in August 2008. It is based on the earlier TLS 1.1 specification. Major differences include:

* The [MD5](https://en.wikipedia.org/wiki/MD5)-[SHA-1](https://en.wikipedia.org/wiki/SHA-1) combination in the [pseudorandom function](https://en.wikipedia.org/wiki/Pseudorandom_function_family) (PRF) was replaced with [SHA-256](https://en.wikipedia.org/wiki/SHA-2), with an option to use [cipher suite](https://en.wikipedia.org/wiki/Cipher_suite) specified PRFs.
* The MD5-SHA-1 combination in the finished message [hash](https://en.wikipedia.org/wiki/Hash_function) was replaced with SHA-256, with an option to use cipher suite specific hash algorithms. However, the size of the hash in the finished message must still be at least 96 [bits](https://en.wikipedia.org/wiki/Bit).[[27]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-27)
* The MD5-SHA-1 combination in the digitally signed element was replaced with a single hash negotiated during [handshake](https://en.wikipedia.org/wiki/Handshaking), which defaults to SHA-1.
* Enhancement in the client's and server's ability to specify which hashes and signature algorithms they accept.
* Expansion of support for [authenticated encryption](https://en.wikipedia.org/wiki/Authenticated_encryption) ciphers, used mainly for [Galois/Counter Mode](https://en.wikipedia.org/wiki/Galois/Counter_Mode) (GCM) and [CCM mode](https://en.wikipedia.org/wiki/CCM_mode) of [Advanced Encryption Standard](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard) (AES) encryption.
* TLS Extensions definition and AES cipher suites were added.[[23]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-urlnvlpubs.nist.gov-23):2

All TLS versions were further refined in [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6176](https://tools.ietf.org/html/rfc6176) in March 2011, removing their backward compatibility with SSL such that TLS sessions never negotiate the use of Secure Sockets Layer (SSL) version 2.0.

**TLS 1.3**

TLS 1.3 was defined in [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [8446](https://tools.ietf.org/html/rfc8446) in August 2018. It is based on the earlier TLS 1.2 specification. Major differences from TLS 1.2 include:[[28]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-WolfSSL,_2019-28)

* Separating key agreement and authentication algorithms from the cipher suites
* Removing support for weak and less-used named [elliptic curves](https://en.wikipedia.org/wiki/Elliptic-curve_cryptography)
* Removing support for MD5 and SHA-224 [cryptographic hash functions](https://en.wikipedia.org/wiki/Cryptographic_hash_function)
* Requiring digital signatures even when a previous configuration is used
* Integrating [HKDF](https://en.wikipedia.org/wiki/HKDF) and the semi-ephemeral DH proposal
* Replacing resumption with [PSK](https://en.wikipedia.org/wiki/TLS-PSK) and tickets
* Supporting 1-[RTT](https://en.wikipedia.org/wiki/Round-trip_delay_time) handshakes and initial support for 0-[RTT](https://en.wikipedia.org/wiki/Round-trip_delay_time)
* Mandating [perfect forward secrecy](https://en.wikipedia.org/wiki/Forward_secrecy), by means of using ephemeral keys during the (EC)DH key agreement
* Dropping support for many insecure or obsolete features including [compression](https://en.wikipedia.org/wiki/Data_compression), renegotiation, non-[AEAD](https://en.wikipedia.org/wiki/Authenticated_encryption) ciphers, non-[PFS](https://en.wikipedia.org/wiki/Forward_secrecy) key exchange (among which are static [RSA](https://en.wikipedia.org/wiki/RSA_(cryptosystem)) and static [DH](https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange) key exchanges), custom [DHE](https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange) groups, EC point format negotiation, Change Cipher Spec protocol, Hello message UNIX time, and the length field AD input to AEAD ciphers
* Prohibiting SSL or RC4 negotiation for backwards compatibility
* Integrating use of session hash
* Deprecating use of the record layer version number and freezing the number for improved backwards compatibility
* Moving some security-related algorithm details from an appendix to the specification and relegating ClientKeyShare to an appendix
* Adding the [ChaCha20](https://en.wikipedia.org/wiki/ChaCha20) stream cipher with the [Poly1305](https://en.wikipedia.org/wiki/Poly1305) message authentication code
* Adding the [Ed25519](https://en.wikipedia.org/wiki/Ed25519) and [Ed448](https://en.wikipedia.org/wiki/Ed448) digital signature algorithms
* Adding the [x25519](https://en.wikipedia.org/wiki/X25519) and [x448](https://en.wikipedia.org/wiki/X448) key exchange protocols
* Adds support for sending multiple [OCSP](https://en.wikipedia.org/wiki/Online_Certificate_Status_Protocol) responses
* Encrypts all handshake messages after the ServerHello

[Network Security Services](https://en.wikipedia.org/wiki/Network_Security_Services) (NSS), the cryptography library developed by [Mozilla](https://en.wikipedia.org/wiki/Mozilla) and used by its web browser [Firefox](https://en.wikipedia.org/wiki/Firefox), enabled TLS 1.3 by default in February 2017.[[29]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-NSS-3.29-29) TLS 1.3 support was subsequently added — but due to compatibility issues for a small number of users, not automatically enabled[[30]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-30) — to [Firefox 52.0](https://en.wikipedia.org/wiki/History_of_Firefox#Firefox_52_through_59), which was released in March 2017. TLS 1.3 was enabled by default in May 2018 with the release of [Firefox 60.0](https://en.wikipedia.org/wiki/History_of_Firefox#Firefox_60_through_67).[[31]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-31)

[Google Chrome](https://en.wikipedia.org/wiki/Google_Chrome) set TLS 1.3 as the default version for a short time in 2017. It then removed it as the default, due to incompatible middleboxes such as [Blue Coat web proxies](https://en.wikipedia.org/wiki/Blue_Coat_Systems).[[32]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-32)

During the IETF 100 [Hackathon](https://en.wikipedia.org/wiki/Hackathon) which took place in [Singapore](https://en.wikipedia.org/wiki/Singapore), The TLS Group worked on adapting [open-source applications](https://en.wikipedia.org/wiki/Open-source_software) to use TLS 1.3.[[33]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-33)[[34]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-:1-34) The TLS group was made up of individuals from [Japan](https://en.wikipedia.org/wiki/Japan), [United Kingdom](https://en.wikipedia.org/wiki/United_Kingdom), and [Mauritius](https://en.wikipedia.org/wiki/Mauritius) via the cyberstorm.mu team.[[34]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-:1-34) During the IETF 101 Hackathon which took place in [London](https://en.wikipedia.org/wiki/London), more work was done on application support of TLS 1.3.[[35]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-35) During IETF 102 Hackathon, work continued to inter-operate lesser known TLS 1.3 implementations along with application integration.[[36]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-36)

[wolfSSL](https://en.wikipedia.org/wiki/WolfSSL) enabled the use of TLS 1.3 as of version 3.11.1, released in May 2017.[[37]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-37) As the first commercial TLS 1.3 implementation, wolfSSL 3.11.1 supported Draft 18 and now supports Draft 28,[[38]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-38) the final version, as well as many older versions. A series of blogs was published on the performance difference between TLS 1.2 and 1.3.[[39]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-39)

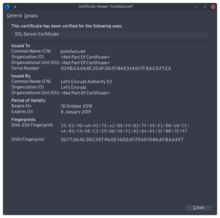
In September 2018, the popular [OpenSSL](https://en.wikipedia.org/wiki/OpenSSL) project released version 1.1.1 of its library, in which support for TLS 1.3 was "[t]he headline new feature".[[40]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-40)

**Enterprise Transport Security**

The [Electronic Frontier Foundation](https://en.wikipedia.org/wiki/Electronic_Frontier_Foundation) praised TLS 1.3 and expressed concern about the variant protocol **Enterprise Transport Security** (**ETS**) that intentionally disables important security measures in TLS 1.3.[[41]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-41) ETS is a published standard known as the **ETSI TS103523-3**, "Middlebox Security Protocol, Part3: Enterprise Transport Security", and intended for use entirely within proprietary networks such as banking systems to enable detection of malware placement, unlawful data exfiltration, and compliance with regulatory auditing mandates.

**Digital certificates**

Main article: [Public key certificate](https://en.wikipedia.org/wiki/Public_key_certificate)

[](https://en.wikipedia.org/wiki/File:Let%27s_Encrypt_Certificate_Firefox_example.png)

Example of a website with digital certificate

A digital certificate certifies the ownership of a public key by the named subject of the certificate, and indicates certain expected usages of that key. This allows others (relying parties) to rely upon signatures or on assertions made by the private key that corresponds to the certified public key.

**Certificate authorities**

Main article: [Certificate authority](https://en.wikipedia.org/wiki/Certificate_authority)

TLS typically relies on a set of trusted third-party certificate authorities to establish the authenticity of certificates. Trust is usually anchored in a list of certificates distributed with user agent software,[[42]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-42) and can be modified by the relying party.

According to [Netcraft](https://en.wikipedia.org/wiki/Netcraft), who monitors active TLS certificates, the market-leading certificate authority (CA) has been [Symantec](https://en.wikipedia.org/wiki/NortonLifeLock) since the beginning of their survey (or [VeriSign](https://en.wikipedia.org/wiki/Verisign) before the authentication services business unit was purchased by Symantec). As of 2015, Symantec accounted for just under a third of all certificates and 44% of the valid certificates used by the 1 million busiest websites, as counted by Netcraft.[[43]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-43) In 2017, Symantec sold its TLS/SSL business to DigiCert.[[44]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-44) In an updated report, it was shown that [IdenTrust](https://en.wikipedia.org/wiki/IdenTrust), [DigiCert](https://en.wikipedia.org/wiki/DigiCert), and [Sectigo](https://en.wikipedia.org/wiki/Sectigo) are the top 3 certificate authorities in terms of market share since May 2019.[[45]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-45)

As a consequence of choosing [X.509](https://en.wikipedia.org/wiki/X.509) certificates, certificate authorities and a [public key infrastructure](https://en.wikipedia.org/wiki/Public_key_infrastructure) are necessary to verify the relation between a certificate and its owner, as well as to generate, sign, and administer the validity of certificates. While this can be more convenient than verifying the identities via a [web of trust](https://en.wikipedia.org/wiki/Web_of_trust), the [2013 mass surveillance disclosures](https://en.wikipedia.org/wiki/Global_surveillance_disclosures_(2013%E2%80%93present)) made it more widely known that certificate authorities are a weak point from a security standpoint, allowing [man-in-the-middle attacks](https://en.wikipedia.org/wiki/Man-in-the-middle_attack) (MITM) if the certificate authority cooperates (or is compromised).[[46]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-46)[[47]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-47)

**Algorithms**

See also: [Cipher suite](https://en.wikipedia.org/wiki/Cipher_suite)

**Key exchange or key agreement**

Before a client and server can begin to exchange information protected by TLS, they must securely exchange or agree upon an encryption key and a cipher to use when encrypting data (see [§ Cipher](https://en.wikipedia.org/wiki/Transport_Layer_Security#Cipher)). Among the methods used for key exchange/agreement are: public and private keys generated with [RSA](https://en.wikipedia.org/wiki/RSA_(algorithm)) (denoted TLS\_RSA in the TLS handshake protocol), [Diffie–Hellman](https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman) (TLS\_DH), ephemeral Diffie–Hellman (TLS\_DHE), [elliptic-curve Diffie–Hellman](https://en.wikipedia.org/wiki/Elliptic-curve_Diffie%E2%80%93Hellman) (TLS\_ECDH), ephemeral elliptic-curve Diffie–Hellman (TLS\_ECDHE), [anonymous Diffie–Hellman](https://en.wikipedia.org/wiki/Key-agreement_protocol#Exponential_key_exchange) (TLS\_DH\_anon),[[2]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RFC5246-2) [pre-shared key](https://en.wikipedia.org/wiki/TLS-PSK) (TLS\_PSK)[[48]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RFC4279-48) and [Secure Remote Password](https://en.wikipedia.org/wiki/TLS-SRP) (TLS\_SRP).[[49]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RFC5054-49)

The TLS\_DH\_anon and TLS\_ECDH\_anon key agreement methods do not authenticate the server or the user and hence are rarely used because those are vulnerable to [man-in-the-middle attacks](https://en.wikipedia.org/wiki/Man-in-the-middle_attack). Only TLS\_DHE and TLS\_ECDHE provide [forward secrecy](https://en.wikipedia.org/wiki/Transport_Layer_Security#Forward_secrecy).

Public key certificates used during exchange/agreement also vary in the size of the public/private encryption keys used during the exchange and hence the robustness of the security provided. In July 2013, [Google](https://en.wikipedia.org/wiki/Google) announced that it would no longer use 1024-bit public keys and would switch instead to 2048-bit keys to increase the security of the TLS encryption it provides to its users because the encryption strength is directly related to the [key size](https://en.wikipedia.org/wiki/Key_size).[[4]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-:0-4)[[50]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-50)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Key exchange/agreement and authentication | | | | | | | |
| **Algorithm** | **SSL 2.0** | **SSL 3.0** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** | **Status** |
| [**RSA**](https://en.wikipedia.org/wiki/RSA_(cryptosystem)) | Yes | Yes | Yes | Yes | Yes | No | Defined for TLS 1.2 in RFCs |
| [**DH**](https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange)**-**[**RSA**](https://en.wikipedia.org/wiki/RSA_(cryptosystem)) | No | Yes | Yes | Yes | Yes | No |
| [**DHE**](https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange)**-**[**RSA**](https://en.wikipedia.org/wiki/RSA_(cryptosystem)) **(**[**forward secrecy**](https://en.wikipedia.org/wiki/Transport_Layer_Security#Forward_secrecy)**)** | No | Yes | Yes | Yes | Yes | Yes |
| [**ECDH**](https://en.wikipedia.org/wiki/Elliptic-curve_Diffie%E2%80%93Hellman)**-**[**RSA**](https://en.wikipedia.org/wiki/RSA_(cryptosystem)) | No | No | Yes | Yes | Yes | No |
| [**ECDHE**](https://en.wikipedia.org/wiki/Elliptic-curve_Diffie%E2%80%93Hellman)**-**[**RSA**](https://en.wikipedia.org/wiki/RSA_(cryptosystem)) **(**[**forward secrecy**](https://en.wikipedia.org/wiki/Transport_Layer_Security#Forward_secrecy)**)** | No | No | Yes | Yes | Yes | Yes |
| [**DH**](https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange)**-**[**DSS**](https://en.wikipedia.org/wiki/Digital_Signature_Algorithm) | No | Yes | Yes | Yes | Yes | No |
| [**DHE**](https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange)**-**[**DSS**](https://en.wikipedia.org/wiki/Digital_Signature_Algorithm) **(**[**forward secrecy**](https://en.wikipedia.org/wiki/Transport_Layer_Security#Forward_secrecy)**)** | No | Yes | Yes | Yes | Yes | No[[51]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-51) |
| [**ECDH**](https://en.wikipedia.org/wiki/Elliptic-curve_Diffie%E2%80%93Hellman)**-**[**ECDSA**](https://en.wikipedia.org/wiki/Elliptic_Curve_DSA) | No | No | Yes | Yes | Yes | No |
| [**ECDHE**](https://en.wikipedia.org/wiki/Elliptic-curve_Diffie%E2%80%93Hellman)**-**[**ECDSA**](https://en.wikipedia.org/wiki/Elliptic_Curve_DSA) **(**[**forward secrecy**](https://en.wikipedia.org/wiki/Transport_Layer_Security#Forward_secrecy)**)** | No | No | Yes | Yes | Yes | Yes |
| [**ECDH**](https://en.wikipedia.org/wiki/ECDH)**-**[**EdDSA**](https://en.wikipedia.org/wiki/EdDSA) | No | No | Yes | Yes | Yes | No |
| [**ECDHE**](https://en.wikipedia.org/wiki/ECDHE)**-**[**EdDSA**](https://en.wikipedia.org/wiki/EdDSA) **(**[**forward secrecy**](https://en.wikipedia.org/wiki/Transport_Layer_Security#Forward_secrecy)**)**[**[52]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-52) | No | No | Yes | Yes | Yes | Yes |
| [**PSK**](https://en.wikipedia.org/wiki/TLS-PSK) | No | No | Yes | Yes | Yes |  |
| [**PSK**](https://en.wikipedia.org/wiki/Pre-shared_key)**-**[**RSA**](https://en.wikipedia.org/wiki/RSA_(cryptosystem)) | No | No | Yes | Yes | Yes |  |
| [**DHE**](https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange)**-**[**PSK**](https://en.wikipedia.org/wiki/Pre-shared_key) **(**[**forward secrecy**](https://en.wikipedia.org/wiki/Transport_Layer_Security#Forward_secrecy)**)** | No | No | Yes | Yes | Yes | Yes |
| [**ECDHE**](https://en.wikipedia.org/wiki/Elliptic-curve_Diffie%E2%80%93Hellman)**-**[**PSK**](https://en.wikipedia.org/wiki/Pre-shared_key) **(**[**forward secrecy**](https://en.wikipedia.org/wiki/Transport_Layer_Security#Forward_secrecy)**)** | No | No | Yes | Yes | Yes | Yes |
| [**SRP**](https://en.wikipedia.org/wiki/TLS-SRP) | No | No | Yes | Yes | Yes |  |
| [**SRP**](https://en.wikipedia.org/wiki/Secure_Remote_Password_protocol)**-**[**DSS**](https://en.wikipedia.org/wiki/Digital_Signature_Algorithm) | No | No | Yes | Yes | Yes |  |
| [**SRP**](https://en.wikipedia.org/wiki/Secure_Remote_Password_protocol)**-**[**RSA**](https://en.wikipedia.org/wiki/RSA_(cryptosystem)) | No | No | Yes | Yes | Yes |  |
| [**Kerberos**](https://en.wikipedia.org/wiki/Kerberos_(protocol)) | No | No | Yes | Yes | Yes |  |
| [**DH**](https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange)**-ANON (insecure)** | No | Yes | Yes | Yes | Yes |  |
| [**ECDH**](https://en.wikipedia.org/wiki/Elliptic-curve_Diffie%E2%80%93Hellman)**-ANON (insecure)** | No | No | Yes | Yes | Yes |  |
| [**GOST R 34.10-94 / 34.10-2001**](https://en.wikipedia.org/wiki/GOST)[**[53]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-gostlink-53) | No | No | Yes | Yes | Yes |  | Proposed in RFC drafts |

**Cipher**

See also: [Cipher suite](https://en.wikipedia.org/wiki/Cipher_suite), [Block cipher](https://en.wikipedia.org/wiki/Block_cipher), and [Cipher security summary](https://en.wikipedia.org/wiki/Cipher_security_summary)

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [Cipher](https://en.wikipedia.org/wiki/Cipher) security against publicly known feasible attacks | | | | | | | | | |
| **Cipher** | | | **Protocol version** | | | | | | **Status** |
| **Type** | **Algorithm** | **Nominal strength (bits)** | **SSL 2.0** | **SSL 3.0** [**[n 1]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-rfc5746-54)[**[n 2]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-renegotiation-55)[**[n 3]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-BEAST-56)[**[n 4]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-POODLEciphertable-57) | **TLS 1.0** [**[n 1]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-rfc5746-54)[**[n 3]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-BEAST-56) | **TLS 1.1** [**[n 1]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-rfc5746-54) | **TLS 1.2** [**[n 1]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-rfc5746-54) | **TLS 1.3** |
| [**Block cipher**](https://en.wikipedia.org/wiki/Block_cipher) **with** [**mode of operation**](https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation) | [**AES**](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard)[**GCM**](https://en.wikipedia.org/wiki/Galois/Counter_Mode)[**[54]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aes-gcm-58)[**[n 5]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aead-59) | 256, 128 | N/A | N/A | N/A | N/A | Secure | Secure | Defined for TLS 1.2 in RFCs |
| [**AES**](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard)[**CCM**](https://en.wikipedia.org/wiki/CCM_mode)[**[55]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aes-ccm-60)[**[n 5]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aead-59) | N/A | N/A | N/A | N/A | Secure | Secure |
| [**AES**](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard)[**CBC**](https://en.wikipedia.org/wiki/Cipher_block_chaining)[**[n 6]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Lucky13-61) | N/A | Insecure | Depends on mitigations | Depends on mitigations | Depends on mitigations | N/A |
| [**Camellia**](https://en.wikipedia.org/wiki/Camellia_(cipher))[**GCM**](https://en.wikipedia.org/wiki/Galois/Counter_Mode)[**[56]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-camellia-gcm-62)[**[n 5]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aead-59) | 256, 128 | N/A | N/A | N/A | N/A | Secure | N/A |
| [**Camellia**](https://en.wikipedia.org/wiki/Camellia_(cipher))[**CBC**](https://en.wikipedia.org/wiki/Cipher_block_chaining)[**[57]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-camellia-cbc-63)[**[n 6]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Lucky13-61) | N/A | Insecure | Depends on mitigations | Depends on mitigations | Depends on mitigations | N/A |
| [**ARIA**](https://en.wikipedia.org/wiki/ARIA_(cipher))[**GCM**](https://en.wikipedia.org/wiki/Galois/Counter_Mode)[**[58]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aria-64)[**[n 5]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aead-59) | 256, 128 | N/A | N/A | N/A | N/A | Secure | N/A |
| [**ARIA**](https://en.wikipedia.org/wiki/ARIA_(cipher))[**CBC**](https://en.wikipedia.org/wiki/Cipher_block_chaining)[**[58]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aria-64)[**[n 6]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Lucky13-61) | N/A | N/A | Depends on mitigations | Depends on mitigations | Depends on mitigations | N/A |
| [**SEED**](https://en.wikipedia.org/wiki/SEED_(cipher))[**CBC**](https://en.wikipedia.org/wiki/Cipher_block_chaining)[**[59]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-seed-cbc-65)[**[n 6]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Lucky13-61) | 128 | N/A | Insecure | Depends on mitigations | Depends on mitigations | Depends on mitigations | N/A |
| [**3DES EDE**](https://en.wikipedia.org/wiki/Triple_DES)[**CBC**](https://en.wikipedia.org/wiki/Cipher_block_chaining)[**[n 6]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Lucky13-61)[**[n 7]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Sweet32-67) | 112[[n 8]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-3des-70) | Insecure | Insecure | Insecure | Insecure | Insecure | N/A |
| [**GOST 28147-89**](https://en.wikipedia.org/wiki/GOST_(block_cipher))[**CNT**](https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation#Counter_(CTR))[**[53]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-gostlink-53)[**[n 7]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Sweet32-67) | 256 | N/A | N/A | Insecure | Insecure | Insecure | N/A | Defined in [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4357](https://tools.ietf.org/html/rfc4357) |
| [**IDEA**](https://en.wikipedia.org/wiki/International_Data_Encryption_Algorithm)[**CBC**](https://en.wikipedia.org/wiki/Cipher_block_chaining)[**[n 6]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Lucky13-61)[**[n 7]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Sweet32-67)[**[n 9]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-removal_from_tls1.2-72) | 128 | Insecure | Insecure | Insecure | Insecure | N/A | N/A | Removed from TLS 1.2 |
| [**DES**](https://en.wikipedia.org/wiki/Data_Encryption_Standard)[**CBC**](https://en.wikipedia.org/wiki/Cipher_block_chaining)[**[n 6]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Lucky13-61)[**[n 7]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Sweet32-67)[**[n 9]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-removal_from_tls1.2-72) | 56 | Insecure | Insecure | Insecure | Insecure | N/A | N/A |
| 40[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-EXPORT-73) | Insecure | Insecure | Insecure | N/A | N/A | N/A | Forbidden in TLS 1.1 and later |
| [**RC2**](https://en.wikipedia.org/wiki/RC2)[**CBC**](https://en.wikipedia.org/wiki/Cipher_block_chaining)[**[n 6]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Lucky13-61)[**[n 7]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Sweet32-67) | 40[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-EXPORT-73) | Insecure | Insecure | Insecure | N/A | N/A | N/A |
| [**Stream cipher**](https://en.wikipedia.org/wiki/Stream_cipher) | [**ChaCha20**](https://en.wikipedia.org/wiki/ChaCha20)**-**[**Poly1305**](https://en.wikipedia.org/wiki/Poly1305)[**[64]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chacha20poly1305-74)[**[n 5]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aead-59) | 256 | N/A | N/A | N/A | N/A | Secure | Secure | Defined for TLS 1.2 in RFCs |
| [**RC4**](https://en.wikipedia.org/wiki/RC4)[**[n 11]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-75) | 128 | Insecure | Insecure | Insecure | Insecure | Insecure | N/A | Prohibited in all versions of TLS by [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7465](https://tools.ietf.org/html/rfc7465) |
| 40[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-EXPORT-73) | Insecure | Insecure | Insecure | N/A | N/A | N/A |
| **None** | **Null**[**[n 12]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-76) | – | Insecure | Insecure | Insecure | Insecure | Insecure | N/A | Defined for TLS 1.2 in RFCs |

Notes

 [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5746](https://tools.ietf.org/html/rfc5746) must be implemented to fix a renegotiation flaw that would otherwise break this protocol.

  If libraries implement fixes listed in [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5746](https://tools.ietf.org/html/rfc5746), this violates the SSL 3.0 specification, which the IETF cannot change unlike TLS. Most current libraries implement the fix and disregard the violation that this causes.

  The [BEAST](https://en.wikipedia.org/wiki/Transport_Layer_Security#BEAST_attack) attack breaks all block ciphers (CBC ciphers) used in SSL 3.0 and TLS 1.0 unless mitigated by the client and/or the server. See [§ Web browsers](https://en.wikipedia.org/wiki/Transport_Layer_Security#Web_browsers).

  The [POODLE](https://en.wikipedia.org/wiki/Transport_Layer_Security#POODLE) attack breaks all block ciphers (CBC ciphers) used in SSL 3.0 unless mitigated by the client and/or the server. See [§ Web browsers](https://en.wikipedia.org/wiki/Transport_Layer_Security#Web_browsers).

  [AEAD](https://en.wikipedia.org/wiki/AEAD_block_cipher_modes_of_operation) ciphers (such as [GCM](https://en.wikipedia.org/wiki/Galois/Counter_Mode) and [CCM](https://en.wikipedia.org/wiki/CCM_mode)) can only be used in TLS 1.2 or later.

  CBC ciphers can be attacked with the [Lucky Thirteen attack](https://en.wikipedia.org/wiki/Lucky_Thirteen_attack) if the library is not written carefully to eliminate timing side channels.

  The Sweet32 attack breaks block ciphers with a block size of 64 bits.[[60]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-sweet32-66)

  Although the key length of 3DES is 168 bits, effective security strength of 3DES is only 112 bits,[[61]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-NIST_SP_800-57-68) which is below the recommended minimum of 128 bits.[[62]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-best-practices-69)

  IDEA and DES have been removed from TLS 1.2.[[63]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-71)

  40-bit strength cipher suites were intentionally designed with reduced key lengths to comply with since-rescinded US regulations forbidding the export of cryptographic software containing certain strong encryption algorithms (see [Export of cryptography from the United States](https://en.wikipedia.org/wiki/Export_of_cryptography_from_the_United_States)). These weak suites are forbidden in TLS 1.1 and later.

  Use of RC4 in all versions of TLS is prohibited by [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7465](https://tools.ietf.org/html/rfc7465) (because [RC4 attacks](https://en.wikipedia.org/wiki/Transport_Layer_Security#RC4_attacks) weaken or break RC4 used in SSL/TLS).

* 1.  Authentication only, no encryption.

**Data integrity**

A [message authentication code](https://en.wikipedia.org/wiki/Message_authentication_code) (MAC) is used for data integrity. [HMAC](https://en.wikipedia.org/wiki/HMAC) is used for [CBC](https://en.wikipedia.org/wiki/Cipher_block_chaining) mode of block ciphers. [Authenticated encryption](https://en.wikipedia.org/wiki/Authenticated_encryption) (AEAD) such as [GCM mode](https://en.wikipedia.org/wiki/Galois/Counter_Mode) and [CCM mode](https://en.wikipedia.org/wiki/CCM_mode) uses AEAD-integrated MAC and doesn't use [HMAC](https://en.wikipedia.org/wiki/HMAC).[[65]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-77) HMAC based [PRF](https://en.wikipedia.org/wiki/Pseudorandom_function_family), or [HKDF](https://en.wikipedia.org/wiki/HKDF) is used for TLS handshake.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Data integrity | | | | | | | |
| **Algorithm** | **SSL 2.0** | **SSL 3.0** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** | **Status** |
| [**HMAC**](https://en.wikipedia.org/wiki/HMAC)**-**[**MD5**](https://en.wikipedia.org/wiki/MD5) | Yes | Yes | Yes | Yes | Yes | No | Defined for TLS 1.2 in RFCs |
| [**HMAC**](https://en.wikipedia.org/wiki/HMAC)**-**[**SHA1**](https://en.wikipedia.org/wiki/SHA-1) | No | Yes | Yes | Yes | Yes | No |
| [**HMAC**](https://en.wikipedia.org/wiki/HMAC)**-**[**SHA256/384**](https://en.wikipedia.org/wiki/SHA-2) | No | No | No | No | Yes | No |
| [**AEAD**](https://en.wikipedia.org/wiki/AEAD_block_cipher_modes_of_operation) | No | No | No | No | Yes | Yes |
| [**GOST 28147-89 IMIT**](https://en.wikipedia.org/wiki/GOST_28147-89)[**[53]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-gostlink-53) | No | No | Yes | Yes | Yes |  | Proposed in RFC drafts |
| [**GOST R 34.11-94**](https://en.wikipedia.org/wiki/GOST_(hash_function))[**[53]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-gostlink-53) | No | No | Yes | Yes | Yes |  |

**Applications and adoption**

In applications design, TLS is usually implemented on top of Transport Layer protocols, encrypting all of the protocol-related data of protocols such as [HTTP](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol), [FTP](https://en.wikipedia.org/wiki/File_Transfer_Protocol), [SMTP](https://en.wikipedia.org/wiki/Simple_Mail_Transfer_Protocol), [NNTP](https://en.wikipedia.org/wiki/Network_News_Transfer_Protocol) and [XMPP](https://en.wikipedia.org/wiki/Extensible_Messaging_and_Presence_Protocol).

Historically, TLS has been used primarily with reliable transport protocols such as the [Transmission Control Protocol](https://en.wikipedia.org/wiki/Transmission_Control_Protocol) (TCP). However, it has also been implemented with datagram-oriented transport protocols, such as the [User Datagram Protocol](https://en.wikipedia.org/wiki/User_Datagram_Protocol) (UDP) and the [Datagram Congestion Control Protocol](https://en.wikipedia.org/wiki/Datagram_Congestion_Control_Protocol) (DCCP), usage of which has been standardized independently using the term [Datagram Transport Layer Security](https://en.wikipedia.org/wiki/Datagram_Transport_Layer_Security) (DTLS).

**Websites**

A primary use of TLS is to secure [World Wide Web](https://en.wikipedia.org/wiki/World_Wide_Web) traffic between a [website](https://en.wikipedia.org/wiki/Website) and a [web browser](https://en.wikipedia.org/wiki/Web_browser) encoded with the HTTP protocol. This use of TLS to secure HTTP traffic constitutes the [HTTPS](https://en.wikipedia.org/wiki/Https) protocol.[[66]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-78)

|  |  |  |
| --- | --- | --- |
| Website protocol support | | |
| **Protocol version** | **Website support**[**[67]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-trustworthy_ssl_pulse-79) | **Security**[**[67]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-trustworthy_ssl_pulse-79)[**[68]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-community.qualys-80) |
| **SSL 2.0** | 1.6% | Insecure |
| **SSL 3.0** | 6.7% | Insecure[[69]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-poodle_pdf-81) |
| **TLS 1.0** | 65.0% | Depends on cipher[[n 1]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ciphers-82) and client mitigations[[n 2]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-mitigations-83) |
| **TLS 1.1** | 75.1% | Depends on cipher[[n 1]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ciphers-82) and client mitigations[[n 2]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-mitigations-83) |
| **TLS 1.2** | 96.0% | Depends on cipher[[n 1]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ciphers-82) and client mitigations[[n 2]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-mitigations-83) |
| **TLS 1.3** | 18.4% | Secure |

Notes

 see [§ Cipher](https://en.wikipedia.org/wiki/Transport_Layer_Security#Cipher) table above

* 1.  see [§ Web browsers](https://en.wikipedia.org/wiki/Transport_Layer_Security#Web_browsers) and [§ Attacks against TLS/SSL](https://en.wikipedia.org/wiki/Transport_Layer_Security#Attacks_against_TLS/SSL) sections

**Web browsers**

Further information: [Comparison of web browsers](https://en.wikipedia.org/wiki/Comparison_of_web_browsers)

As of April 2016, the latest versions of all major web browsers support TLS 1.0, 1.1, and 1.2, and have them enabled by default. However, not all supported [Microsoft operating systems](https://en.wikipedia.org/wiki/List_of_Microsoft_operating_systems) support the latest version of IE. Additionally, many operating systems currently support multiple versions of IE, but this has changed according to Microsoft's [Internet Explorer Support Lifecycle Policy FAQ](https://support.microsoft.com/en-us/gp/microsoft-internet-explorer), "beginning January 12, 2016, only the most current version of Internet Explorer available for a supported operating system will receive technical support and security updates." The page then goes on to list the latest supported version of IE at that date for each operating system. The next critical date would be when an operating system reaches the end of life stage, which is in Microsoft's [Windows lifecycle fact sheet](https://windows.microsoft.com/en-us/windows/lifecycle).

Mitigations against known attacks are not enough yet:

* Mitigations against [POODLE attack](https://en.wikipedia.org/wiki/Transport_Layer_Security#POODLE_attack): some browsers already prevent fallback to SSL 3.0; however, this mitigation needs to be supported by not only clients but also servers. Disabling SSL 3.0 itself, implementation of "anti-POODLE record splitting", or denying CBC ciphers in SSL 3.0 is required.
  + Google Chrome: complete (TLS\_FALLBACK\_SCSV is implemented since version 33, fallback to SSL 3.0 is disabled since version 39, SSL 3.0 itself is disabled by default since version 40. Support of SSL 3.0 itself was dropped since version 44.)
  + Mozilla Firefox: complete (support of SSL 3.0 itself is dropped since [version 39](https://en.wikipedia.org/wiki/History_of_Firefox#Version_38–44). SSL 3.0 itself is disabled by default and fallback to SSL 3.0 are disabled since [version 34](https://en.wikipedia.org/wiki/History_of_Firefox#Version_31–37), TLS\_FALLBACK\_SCSV is implemented since version 35. In ESR, SSL 3.0 itself is disabled by default and TLS\_FALLBACK\_SCSV is implemented since ESR 31.3.)
  + Internet Explorer: partial (only in version 11, SSL 3.0 is disabled by default since April 2015. Version 10 and older are still vulnerable against POODLE.)
  + [Opera](https://en.wikipedia.org/wiki/Opera_(web_browser)): complete (TLS\_FALLBACK\_SCSV is implemented since version 20, "anti-POODLE record splitting", which is effective only with client-side implementation, is implemented since version 25, SSL 3.0 itself is disabled by default since version 27. Support of SSL 3.0 itself will be dropped since version 31.)
  + Safari: complete (only on OS X 10.8 and later and iOS 8, CBC ciphers during fallback to SSL 3.0 is denied, but this means it will use RC4, which is not recommended as well. Support of SSL 3.0 itself is dropped on OS X 10.11 and later and iOS 9.)
* Mitigation against [RC4 attacks](https://en.wikipedia.org/wiki/Transport_Layer_Security#RC4_attacks):
  + Google Chrome disabled RC4 except as a fallback since version 43. RC4 is disabled since Chrome 48.
  + Firefox disabled RC4 except as a fallback since version 36. Firefox 44 disabled RC4 by default.
  + Opera disabled RC4 except as a fallback since version 30. RC4 is disabled since Opera 35.
  + Internet Explorer for [Windows 7](https://en.wikipedia.org/wiki/Windows_7) / Server 2008 R2 and for [Windows 8](https://en.wikipedia.org/wiki/Windows_8) / Server 2012 have set the priority of RC4 to lowest and can also disable RC4 except as a fallback through registry settings. Internet Explorer 11 Mobile 11 for [Windows Phone 8.1](https://en.wikipedia.org/wiki/Windows_Phone_8.1) disable RC4 except as a fallback if no other enabled algorithm works. Edge and IE 11 disable RC4 completely in August 2016.
* Mitigation against [FREAK attack](https://en.wikipedia.org/wiki/Transport_Layer_Security#FREAK):
  + The Android Browser included with [Android 4.0](https://en.wikipedia.org/wiki/Android_Ice_Cream_Sandwich) and older is still vulnerable to the FREAK attack.
  + Internet Explorer 11 Mobile is still vulnerable to the FREAK attack.
  + Google Chrome, Internet Explorer (desktop), Safari (desktop & mobile), and Opera (mobile) have FREAK mitigations in place.
  + Mozilla Firefox on all platforms and Google Chrome on Windows were not affected by FREAK.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| TLS/SSL support history of web browsers | | | | | | | | | | | | | | | | | | | |
| **Browser** | **Version** | | **Platforms** | **SSL protocols** | | **TLS protocols** | | | | **Certificate support** | | | **Vulnerabilities fixed**[**[n 1]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-sec-84) | | | | | | **Protocol selection by user** [**[n 2]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-poodle-85) |
| **SSL 2.0 (insecure)** | **SSL 3.0 (insecure)** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** | [**EV**](https://en.wikipedia.org/wiki/Extended_Validation_Certificate)[**[n 3]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-evssl-86)[**[70]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-EVSSLbrowser-87) | [**SHA-2**](https://en.wikipedia.org/wiki/SHA-2)[**[71]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SHA-2-88) | [**ECDSA**](https://en.wikipedia.org/wiki/ECDSA)[**[72]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | **BEAST**[**[n 4]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-BEASTmitigation-90) | **CRIME**[**[n 5]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-CRIMEmitigation-91) | **POODLE (SSLv3)**[**[n 6]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-POODLEmitigation-92) | **RC4**[**[n 7]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-RC4mitigation-93) | **FREAK**[**[73]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-FREAK-1-94)[**[74]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-FREAK-2-95) | **Logjam** |
| [**Google Chrome**](https://en.wikipedia.org/wiki/Google_Chrome) **(**[**Chrome for Android**](https://en.wikipedia.org/wiki/Google_Chrome_for_Android)**)** [**[n 8]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-a-99)[**[n 9]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-b-101) | 1–9 | | [Windows (7+)](https://en.wikipedia.org/wiki/Microsoft_Windows) [macOS (10.10+)](https://en.wikipedia.org/wiki/MacOS) [Linux](https://en.wikipedia.org/wiki/Linux) [Android (4.4+)](https://en.wikipedia.org/wiki/Android_(operating_system)) [iOS (10.0+)](https://en.wikipedia.org/wiki/IOS) [Chrome OS](https://en.wikipedia.org/wiki/Chrome_OS) | Disabled by default | Enabled by default | Yes | No | No | No | Yes (only desktop) | needs SHA-2 compatible OS[[71]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SHA-2-88) | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected [[79]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ChromeBEAST-102) | Vulnerable (HTTPS) | Vulnerable | Vulnerable | Vulnerable (except Windows) | Vulnerable | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 10–20 | | No[[80]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-build.chromium.org-104) | Enabled by default | Yes | No | No | No | Yes (only desktop) | needs SHA-2 compatible OS[[71]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SHA-2-88) | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Vulnerable (HTTPS/SPDY) | Vulnerable | Vulnerable | Vulnerable (except Windows) | Vulnerable | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 21 | | No | Enabled by default | Yes | No | No | No | Yes (only desktop) | needs SHA-2 compatible OS[[71]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SHA-2-88) | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated [[81]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-CRIME-Chrome=Firefox-105) | Vulnerable | Vulnerable | Vulnerable (except Windows) | Vulnerable | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 22–29 | | No | Enabled by default | Yes | Yes[[82]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SSL/TLS_Overview-106) | No[[82]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SSL/TLS_Overview-106)[[83]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Chromium_Issue_90392-107)[[84]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Issue_23503030_Merge_219882-108)[[85]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Issue_278370-109) | No | Yes (only desktop) | needs SHA-2 compatible OS[[71]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SHA-2-88) | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Vulnerable | Vulnerable | Vulnerable (except Windows) | Vulnerable | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 30–32 | | No | Enabled by default | Yes | Yes | Yes​[[83]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Chromium_Issue_90392-107)[[84]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Issue_23503030_Merge_219882-108)[[85]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Issue_278370-109) | No | Yes (only desktop) | needs SHA-2 compatible OS[[71]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SHA-2-88) | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Vulnerable | Vulnerable | Vulnerable (except Windows) | Vulnerable | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 33–37 | | No | Enabled by default | Yes | Yes | Yes | No | Yes (only desktop) | needs SHA-2 compatible OS[[71]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SHA-2-88) | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Partly mitigated [[n 12]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-Chrome33-POODLE-113) | Lowest priority [[88]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Chrome33RC4-114)[[89]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Chrome33changelog-115)[[90]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Issue318442-116) | Vulnerable (except Windows) | Vulnerable | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 38, 39 | | No | Enabled by default | Yes | Yes | Yes | No | Yes (only desktop) | Yes | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Partly mitigated | Lowest priority | Vulnerable (except Windows) | Vulnerable | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 40 | | No | Disabled by default​[[87]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chrome-poodle-112)[[91]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chromium693963003-117) | Yes | Yes | Yes | No | Yes (only desktop) | Yes | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Mitigated [[n 13]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-Chrome40-POODLE-118) | Lowest priority | Vulnerable (except Windows) | Vulnerable | Yes[[n 14]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chromeflags-119) |
| 41, 42 | | No | Disabled by default | Yes | Yes | Yes | No | Yes (only desktop) | Yes | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Mitigated | Lowest priority | Mitigated | Vulnerable | Yes[[n 14]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chromeflags-119) |
| 43 | | No | Disabled by default | Yes | Yes | Yes | No | Yes (only desktop) | Yes | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Mitigated | Only as fallback [[n 15]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4fallback-120)[[92]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chromeissue375342-121) | Mitigated | Vulnerable | Yes[[n 14]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chromeflags-119) |
| 44–47 | | No | No[[93]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chromeissue436391-122) | Yes | Yes | Yes | No | Yes (only desktop) | Yes | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Not affected | Only as fallback [[n 15]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4fallback-120) | Mitigated | Mitigated​[[94]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chromeissue490240-123) | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 48, 49 | | No | No | Yes | Yes | Yes | No | Yes (only desktop) | Yes | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124)[[95]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-1-125)[[96]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-2-126) | Mitigated | Mitigated | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 50–53 | | No | No | Yes | Yes | Yes | No | Yes (only desktop) | Yes | Yes | Not affected | Mitigated | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124)[[95]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-1-125)[[96]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-2-126) | Mitigated | Mitigated | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 54–66 | | No | No | Yes | Yes | Yes | Disabled by default (draft version) | Yes (only desktop) | Yes | Yes | Not affected | Mitigated | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124)[[95]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-1-125)[[96]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-2-126) | Mitigated | Mitigated | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 67–69 | | No | No | Yes | Yes | Yes | Yes (draft version) | Yes (only desktop) | Yes | Yes | Not affected | Mitigated | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124)[[95]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-1-125)[[96]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-2-126) | Mitigated | Mitigated | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 70–83 | | No | No | Yes | Yes | Yes | Yes | Yes (only desktop) | Yes | Yes | Not affected | Mitigated | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124)[[95]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-1-125)[[96]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-2-126) | Mitigated | Mitigated | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 84 | | No | No | Warn by default | Warn by default | Yes | Yes | Yes (only desktop) | Yes | Yes | Not affected | Mitigated | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124)[[95]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-1-125)[[96]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-2-126) | Mitigated | Mitigated | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| **Browser** | **Version** | | **Platforms** | **SSL 2.0 (insecure)** | **SSL 3.0 (insecure)** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** | **EV certificate** | **SHA-2 certificate** | **ECDSA certificate** | **BEAST** | **CRIME** | **POODLE (SSLv3)** | **RC4** | **FREAK** | **Logjam** | **Protocol selection by user** |
| [**Microsoft Edge**](https://en.wikipedia.org/wiki/Microsoft_Edge)[**(Chromium based)**](https://en.wikipedia.org/wiki/Chromium_(web_browser)) **OS independent** | 79–83 | | [**Windows (7+)**](https://en.wikipedia.org/wiki/Microsoft_Windows)[**macOS (10.12+)**](https://en.wikipedia.org/wiki/MacOS)[**Linux**](https://en.wikipedia.org/wiki/Linux)[**Android (4.4+)**](https://en.wikipedia.org/wiki/Android_(operating_system))[**iOS (10.0+)**](https://en.wikipedia.org/wiki/IOS) | No | No | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Mitigated | Not affected | Not affected | Disabled by default | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 84 | | No | No | Warn by default | Warn by default | Yes | Yes | Yes | Yes | Yes | Mitigated | Not affected | Not affected | Disabled by default | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 88[[97]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-127) | | No | No | No | No | Yes | Yes | Yes | Yes | Yes | Mitigated | Not affected | Not affected | Disabled by default | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| **Browser** | **Version** | | **Platforms** | **SSL 2.0 (insecure)** | **SSL 3.0 (insecure)** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** | **EV certificate** | **SHA-2 certificate** | **ECDSA certificate** | **BEAST** | **CRIME** | **POODLE (SSLv3)** | **RC4** | **FREAK** | **Logjam** | **Protocol selection by user** |
| [**Mozilla Firefox**](https://en.wikipedia.org/wiki/Firefox) **(**[**Firefox for mobile**](https://en.wikipedia.org/wiki/Firefox_for_mobile)**)** [**[n 17]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-c-128) | 1.0, 1.5 | | [Windows (7+)](https://en.wikipedia.org/wiki/Microsoft_Windows) [macOS (10.9+)](https://en.wikipedia.org/wiki/MacOS) [Linux](https://en.wikipedia.org/wiki/Linux) [Android (4.1+)](https://en.wikipedia.org/wiki/Android_(OS)) [iOS (10.3+)](https://en.wikipedia.org/wiki/IOS) [~~Firefox OS~~](https://en.wikipedia.org/wiki/Firefox_OS) [~~Maemo~~](https://en.wikipedia.org/wiki/Maemo)  ESR only for: [Windows (7+)](https://en.wikipedia.org/wiki/Microsoft_Windows) [macOS (10.9+)](https://en.wikipedia.org/wiki/MacOS) [Linux](https://en.wikipedia.org/wiki/Linux) | Enabled by default [[98]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SecurityinFirefox-129) | Enabled by default [[98]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SecurityinFirefox-129) | Yes[[98]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SecurityinFirefox-129) | No | No | No | No | Yes[[71]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SHA-2-88) | No | Not affected [[99]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-FirefoxBEAST-130) | Not affected | Vulnerable | Vulnerable | Not affected | Vulnerable | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 2 | | Disabled by default [[98]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SecurityinFirefox-129)[[100]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-FxSSL2-131) | Enabled by default | Yes | No | No | No | No | Yes | Yes[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Not affected | Vulnerable | Vulnerable | Not affected | Vulnerable | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 3–7 | | Disabled by default | Enabled by default | Yes | No | No | No | Yes | Yes | Yes | Not affected | Not affected | Vulnerable | Vulnerable | Not affected | Vulnerable | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 8–10 ESR 10 | | No[[100]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-FxSSL2-131) | Enabled by default | Yes | No | No | No | Yes | Yes | Yes | Not affected | Not affected | Vulnerable | Vulnerable | Not affected | Vulnerable | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11–14 | | No | Enabled by default | Yes | No | No | No | Yes | Yes | Yes | Not affected | Vulnerable (SPDY)[[81]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-CRIME-Chrome=Firefox-105) | Vulnerable | Vulnerable | Not affected | Vulnerable | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 15–22 ESR 17.0–17.0.10 | | No | Enabled by default | Yes | No | No | No | Yes | Yes | Yes | Not affected | Mitigated | Vulnerable | Vulnerable | Not affected | Vulnerable | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| ESR 17.0.11 | | No | Enabled by default | Yes | No | No | No | Yes | Yes | Yes | Not affected | Mitigated | Vulnerable | Lowest priority [[101]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-NSS-3.15.3-132)[[102]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-mfsa2013-103-133) | Not affected | Vulnerable | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 23 | | No | Enabled by default | Yes | Disabled by default [[103]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Bug565047-134) | No | No | Yes | Yes | Yes | Not affected | Mitigated | Vulnerable | Vulnerable | Not affected | Vulnerable | Yes[[n 18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aboutconfig-135) |
| 24, 25.0.0 ESR 24.0–24.1.0 | | No | Enabled by default | Yes | Disabled by default | Disabled by default [[104]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Bug480514-136) | No | Yes | Yes | Yes | Not affected | Mitigated | Vulnerable | Vulnerable | Not affected | Vulnerable | Yes[[n 18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aboutconfig-135) |
| 25.0.1, 26 ESR 24.1.1 | | No | Enabled by default | Yes | Disabled by default | Disabled by default | No | Yes | Yes | Yes | Not affected | Mitigated | Vulnerable | Lowest priority [[101]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-NSS-3.15.3-132)[[102]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-mfsa2013-103-133) | Not affected | Vulnerable | Yes[[n 18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aboutconfig-135) |
| 27–33 ESR 31.0–31.2 | | No | Enabled by default | Yes | Yes​[[105]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Bug733647-137)[[106]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Fx27-138) | Yes​[[107]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Bug861266-139)[[106]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Fx27-138) | No | Yes | Yes | Yes | Not affected | Mitigated | Vulnerable | Lowest priority | Not affected | Vulnerable | Yes[[n 18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aboutconfig-135) |
| 34, 35 ESR 31.3–31.7 | | No | Disabled by default [[108]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-firefox-poodle-140)[[109]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Fx34-141) | Yes | Yes | Yes | No | Yes | Yes | Yes | Not affected | Mitigated | Mitigated [[n 19]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Fx-POODLE-144) | Lowest priority | Not affected | Vulnerable | Yes[[n 18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aboutconfig-135) |
| ESR 31.8 | | No | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Not affected | Mitigated | Mitigated | Lowest priority | Not affected | Mitigated​[[112]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-FxLogjam-145) | Yes[[n 18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aboutconfig-135) |
| 36–38 ESR 38.0 | | No | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Not affected | Mitigated | Mitigated | Only as fallback [[n 15]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4fallback-120)[[113]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-https-146) | Not affected | Vulnerable | Yes[[n 18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aboutconfig-135) |
| ESR 38.1–38.8 | | No | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Not affected | Mitigated | Mitigated | Only as fallback [[n 15]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4fallback-120) | Not affected | Mitigated​[[112]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-FxLogjam-145) | Yes[[n 18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aboutconfig-135) |
| 39–43 | | No | No[[114]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Fx39-147) | Yes | Yes | Yes | No | Yes | Yes | Yes | Not affected | Mitigated | Not affected | Only as fallback [[n 15]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4fallback-120) | Not affected | Mitigated​[[112]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-FxLogjam-145) | Yes[[n 18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aboutconfig-135) |
| 44–48  ESR 45 | | No | No | Yes | Yes | Yes | No | Yes | Yes | Yes | Not affected | Mitigated | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124)[[115]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4Fx-148)[[116]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4Fx-2-149)[[117]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4Fx-3-150)[[118]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Fx44-151) | Not affected | Mitigated | Yes[[n 18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aboutconfig-135) |
| 49–59 ESR 52 | | No | No | Yes | Yes | Yes | Disabled by default (draft version)​[[119]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-152) | Yes | Yes | Yes | Not affected | Mitigated | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Not affected | Mitigated | Yes[[n 18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aboutconfig-135) |
| 60–62 ESR 60 | | No | No | Yes | Yes | Yes | Yes (draft version) | Yes | Yes | Yes | Not affected | Mitigated | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Not affected | Mitigated | Yes[[n 18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aboutconfig-135) |
| 63–77 ESR 68.0–68.9 | | No | No | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Not affected | Mitigated | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Not affected | Mitigated | Yes[[n 18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aboutconfig-135) |
| ESR 68.10 | |
| ESR 78.0 | 78 | No | No | Disabled by default[[120]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Firefox78-RN-153) | Disabled by default[[120]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Firefox78-RN-153) | Yes | Yes | Yes | Yes | Yes | Not affected | Mitigated | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Not affected | Mitigated | Yes[[n 18]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-aboutconfig-135) |
| **Browser** | **Version** | | **Platforms** | **SSL 2.0 (insecure)** | **SSL 3.0 (insecure)** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** | **EV certificate** | **SHA-2 certificate** | **ECDSA certificate** | **BEAST** | **CRIME** | **POODLE (SSLv3)** | **RC4** | **FREAK** | **Logjam** | **Protocol selection by user** |
| [**Opera Browser**](https://en.wikipedia.org/wiki/Opera_(web_browser)) **(**[**Opera Mobile**](https://en.wikipedia.org/wiki/Opera_Mobile)**) (**[**Pre-Presto and Presto**](https://en.wikipedia.org/wiki/Presto_(layout_engine))**)** [**[n 20]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-e-155) | 1–2 | | [~~Windows~~](https://en.wikipedia.org/wiki/Microsoft_Windows) [~~macOS~~](https://en.wikipedia.org/wiki/MacOS) [~~Linux~~](https://en.wikipedia.org/wiki/Linux) [~~Android~~](https://en.wikipedia.org/wiki/Android_(OS)) [~~Symbian S60~~](https://en.wikipedia.org/wiki/S60_(software_platform)) [~~Maemo~~](https://en.wikipedia.org/wiki/Maemo) [~~Windows Mobile~~](https://en.wikipedia.org/wiki/Windows_Mobile) | No SSL/TLS support[[122]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Opera2-156) | | | | | | | | | | | | | | | |
| 3 | | Yes[[123]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Opera3-157) | No | No | No | No | No | No | No | No | No SSL 3.0 or TLS support | | | Vulnerable | Unknown | Unknown | N/A |
| 4 | | Yes | Yes[[124]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Opera4-158) | No | No | No | No | No | No | No | Vulnerable | Not affected | Vulnerable | Vulnerable | Unknown | Unknown | Unknown |
| 5 | | Enabled by default | Enabled by default | Yes[[125]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Opera5-159) | No | No | No | No | No | No | Vulnerable | Not affected | Vulnerable | Vulnerable | Unknown | Unknown | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 6–7 | | Enabled by default | Enabled by default | Yes[[125]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Opera5-159) | No | No | No | No | Yes[[71]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SHA-2-88) | No | Vulnerable | Not affected | Vulnerable | Vulnerable | Unknown | Unknown | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 8 | | Enabled by default | Enabled by default | Yes | Disabled by default [[126]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-opera.com-160) | No | No | No | Yes | No | Vulnerable | Not affected | Vulnerable | Vulnerable | Unknown | Unknown | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 9 | | Disabled by default [[127]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-161) | Enabled by default | Yes | Yes | No | No | since v9.5 (only desktop) | Yes | No | Vulnerable | Not affected | Vulnerable | Vulnerable | Unknown | Unknown | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 10–11.52 | | No[[128]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Opera10b-162) | Enabled by default | Yes | Disabled by default | Disabled by default [[128]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Opera10b-162) | No | Yes (only desktop) | Yes | No | Vulnerable | Not affected | Vulnerable | Vulnerable | Unknown | Unknown | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11.60–11.64 | | No | Enabled by default | Yes | Disabled by default | Disabled by default | No | Yes (only desktop) | Yes | No | Mitigated [[129]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-my.opera.com-163) | Not affected | Vulnerable | Vulnerable | Unknown | Unknown | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 12–12.14 | | No | Disabled by default [[n 21]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-Opera12-POODLE-164) | Yes | Disabled by default | Disabled by default | No | Yes (only desktop) | Yes | No | Mitigated | Not affected | Mitigated [[n 21]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-Opera12-POODLE-164) | Vulnerable | Unknown | Mitigated​[[131]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-OperaLogjam-166) | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 12.15–12.17 | | No | Disabled by default | Yes | Disabled by default | Disabled by default | No | Yes (only desktop) | Yes | No | Mitigated | Not affected | Mitigated | Partly mitigated [[132]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ReferenceA-167)[[133]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-On_the_Precariousness_of_RC4-168) | Unknown | Mitigated​[[131]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-OperaLogjam-166) | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 12.18 | | No | Disabled by default | Yes | Yes[[134]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-opera1218-169) | Yes[[134]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-opera1218-169) | No | Yes (only desktop) | Yes | Yes[[134]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-opera1218-169) | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124)[[134]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-opera1218-169) | Mitigated​[[134]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-opera1218-169) | Mitigated​[[131]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-OperaLogjam-166) | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| **Browser** | **Version** | | **Platforms** | **SSL 2.0 (insecure)** | **SSL 3.0 (insecure)** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** | **EV certificate** | **SHA-2 certificate** | **ECDSA certificate** | **BEAST** | **CRIME** | **POODLE (SSLv3)** | **RC4** | **FREAK** | **Logjam** | **Protocol selection by user** |
| [**Opera Browser**](https://en.wikipedia.org/wiki/Opera_(web_browser)) **(**[**Opera Mobile**](https://en.wikipedia.org/wiki/Opera_Mobile)**) (**[**Webkit**](https://en.wikipedia.org/wiki/Webkit) **and** [**Blink**](https://en.wikipedia.org/wiki/Blink_(layout_engine))**)** [**[n 22]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-f-172) | 14–16 | | [Windows (7+)](https://en.wikipedia.org/wiki/Microsoft_Windows) [macOS (10.11+)](https://en.wikipedia.org/wiki/MacOS) [Linux](https://en.wikipedia.org/wiki/Linux) [Android (4.4+)](https://en.wikipedia.org/wiki/Android_(OS)) | No | Enabled by default | Yes | Yes[[137]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-opera16-173) | No[[137]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-opera16-173) | No | Yes (only desktop) | needs SHA-2 compatible OS[[71]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SHA-2-88) | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Vulnerable | Vulnerable | Vulnerable (except Windows) | Vulnerable | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 17–19 | | No | Enabled by default | Yes | Yes[[138]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-opera17-174) | Yes[[138]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-opera17-174) | No | Yes (only desktop) | needs SHA-2 compatible OS[[71]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SHA-2-88) | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Vulnerable | Vulnerable | Vulnerable (except Windows) | Vulnerable | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 20–24 | | No | Enabled by default | Yes | Yes | Yes | No | Yes (only desktop) | needs SHA-2 compatible OS[[71]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SHA-2-88) | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Partly mitigated [[n 23]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-Opera20-POODLE-176) | Lowest priority [[139]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-opera20-175) | Vulnerable (except Windows) | Vulnerable | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 25, 26 | | No | Enabled by default [[n 24]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Opera25-SSL3.0-177) | Yes | Yes | Yes | No | Yes (only desktop) | Yes | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Mitigated [[n 25]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-opera25-178) | Lowest priority | Vulnerable (except Windows) | Vulnerable | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 27 | | No | Disabled by default [[91]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chromium693963003-117) | Yes | Yes | Yes | No | Yes (only desktop) | Yes | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Mitigated [[n 26]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-opera27-179) | Lowest priority | Vulnerable (except Windows) | Vulnerable | Yes[[n 27]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-operaflags-180) (only desktop) |
| 28, 29 | | No | Disabled by default | Yes | Yes | Yes | No | Yes (only desktop) | Yes | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Mitigated | Lowest priority | Mitigated | Vulnerable | Yes[[n 27]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-operaflags-180) (only desktop) |
| 30 | | No | Disabled by default | Yes | Yes | Yes | No | Yes (only desktop) | Yes | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Mitigated | Only as fallback [[n 15]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4fallback-120)[[92]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chromeissue375342-121) | Mitigated | Mitigated​[[131]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-OperaLogjam-166) | Yes[[n 27]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-operaflags-180) (only desktop) |
| 31–34 | | No | No[[93]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chromeissue436391-122) | Yes | Yes | Yes | No | Yes (only desktop) | Yes | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Not affected | Only as fallback [[n 15]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4fallback-120)[[92]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chromeissue375342-121) | Mitigated | Mitigated | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 35, 36 | | No | No | Yes | Yes | Yes | No | Yes (only desktop) | Yes | needs ECC compatible OS[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Not affected | Mitigated | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124)[[95]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-1-125)[[96]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-2-126) | Mitigated | Mitigated | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 37–40 | | No | No | Yes | Yes | Yes | No | Yes (only desktop) | Yes | Yes | Not affected | Mitigated | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124)[[95]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-1-125)[[96]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-2-126) | Mitigated | Mitigated | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 41–56 | | No | No | Yes | Yes | Yes | Disabled by default (draft version) | Yes (only desktop) | Yes | Yes | Not affected | Mitigated | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124)[[95]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-1-125)[[96]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-2-126) | Mitigated | Mitigated | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| 57–68 | 69 | No | No | Yes | Yes | Yes | Yes | Yes (only desktop) | Yes | Yes | Not affected | Mitigated | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124)[[95]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-1-125)[[96]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-Chrome48-2-126) | Mitigated | Mitigated | Temporary [[n 11]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viacommandline-110) |
| **Browser** | **Version** | | **Platforms** | **SSL 2.0 (insecure)** | **SSL 3.0 (insecure)** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** | **EV certificate** | **SHA-2 certificate** | **ECDSA certificate** | **BEAST** | **CRIME** | **POODLE (SSLv3)** | **RC4** | **FREAK** | **Logjam** | **Protocol selection by user** |
| [**Microsoft Internet Explorer**](https://en.wikipedia.org/wiki/Internet_Explorer) **(1–10)** [**[n 28]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-d-183) | [1.x](https://en.wikipedia.org/wiki/Internet_Explorer_1) | | Windows [3.1](https://en.wikipedia.org/wiki/Windows_3.1x), [95](https://en.wikipedia.org/wiki/Windows_95), [NT](https://en.wikipedia.org/wiki/Windows_NT),[[n 29]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-oldieonnt-184)[[n 30]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-k-185) Mac OS [7](https://en.wikipedia.org/wiki/System_7), [8](https://en.wikipedia.org/wiki/Mac_OS_8) | No SSL/TLS support | | | | | | | | | | | | | | | |
| [2](https://en.wikipedia.org/wiki/Internet_Explorer_2) | | Yes | No | No | No | No | No | No | No | No | No SSL 3.0 or TLS support | | | Vulnerable | Vulnerable | Vulnerable | N/A |
| [3](https://en.wikipedia.org/wiki/Internet_Explorer_3) | | Yes | Yes[[142]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-IESSL3TLS1-186) | No | No | No | No | No | No | No | Vulnerable | Not affected | Vulnerable | Vulnerable | Vulnerable | Vulnerable | Unknown |
| [4](https://en.wikipedia.org/wiki/Internet_Explorer_4), [5](https://en.wikipedia.org/wiki/Internet_Explorer_5), [6](https://en.wikipedia.org/wiki/Internet_Explorer_6) | | Windows [3.1](https://en.wikipedia.org/wiki/Windows_3.1x), [95](https://en.wikipedia.org/wiki/Windows_95), [98](https://en.wikipedia.org/wiki/Windows_98), [NT](https://en.wikipedia.org/wiki/Windows_NT), [2000](https://en.wikipedia.org/wiki/Windows_2000)[[n 29]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-oldieonnt-184)[[n 30]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-k-185) Mac OS [7.1](https://en.wikipedia.org/wiki/System_7), [8](https://en.wikipedia.org/wiki/Mac_OS_8), [X](https://en.wikipedia.org/wiki/Mac_OS_X), [Solaris](https://en.wikipedia.org/wiki/Solaris_(operating_system)), [HP-UX](https://en.wikipedia.org/wiki/HP-UX) | Enabled by default | Enabled by default | Disabled by default [[142]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-IESSL3TLS1-186) | No | No | No | No | No | No | Vulnerable | Not affected | Vulnerable | Vulnerable | Vulnerable | Vulnerable | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| [6](https://en.wikipedia.org/wiki/Internet_Explorer_6) | | [Windows XP](https://en.wikipedia.org/wiki/Windows_XP)[[n 30]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-k-185) | Enabled by default | Enabled by default | Disabled by default | No | No | No | No | Yes [[n 31]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Win2003SHA2-187)[[143]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-WinXP2003SHA2-188) | No | Mitigated | Not affected | Vulnerable | Vulnerable | Vulnerable | Vulnerable | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| [7](https://en.wikipedia.org/wiki/Internet_Explorer_7), [8](https://en.wikipedia.org/wiki/Internet_Explorer_8) | | Disabled by default [[144]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-IE7HTTPS-189) | Enabled by default | Yes[[144]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-IE7HTTPS-189) | No | No | No | Yes | Yes [[n 31]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Win2003SHA2-187)[[143]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-WinXP2003SHA2-188) | No | Mitigated | Not affected | Vulnerable | Vulnerable | Vulnerable | Vulnerable | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| [6](https://en.wikipedia.org/wiki/Internet_Explorer_6) | | [Server 2003](https://en.wikipedia.org/wiki/Windows_Server_2003)[[n 30]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-k-185) | Enabled by default | Enabled by default | Disabled by default | No | No | No | No | Yes [[n 31]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Win2003SHA2-187)[[143]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-WinXP2003SHA2-188) | No | Mitigated | Not affected | Vulnerable | Vulnerable | Mitigated [[147]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-031-192) | Mitigated [[148]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-055-193) | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| [7](https://en.wikipedia.org/wiki/Internet_Explorer_7), [8](https://en.wikipedia.org/wiki/Internet_Explorer_8) | | Disabled by default [[144]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-IE7HTTPS-189) | Enabled by default | Yes[[144]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-IE7HTTPS-189) | No | No | No | Yes | Yes [[n 31]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Win2003SHA2-187)[[143]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-WinXP2003SHA2-188) | No | Mitigated | Not affected | Vulnerable | Vulnerable | Mitigated [[147]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-031-192) | Mitigated [[148]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-055-193) | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| [7](https://en.wikipedia.org/wiki/Internet_Explorer_7), [8](https://en.wikipedia.org/wiki/Internet_Explorer_8), [9](https://en.wikipedia.org/wiki/Internet_Explorer_9) | | [Windows Vista](https://en.wikipedia.org/wiki/Windows_Vista) | Disabled by default | Enabled by default | Yes | No | No | No | Yes | Yes | Yes[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Mitigated | Not affected | Vulnerable | Vulnerable | Mitigated [[147]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-031-192) | Mitigated [[148]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-055-193) | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| [7](https://en.wikipedia.org/wiki/Internet_Explorer_7), [8](https://en.wikipedia.org/wiki/Internet_Explorer_8), [9](https://en.wikipedia.org/wiki/Internet_Explorer_9) | | [Server 2008](https://en.wikipedia.org/wiki/Windows_Server_2008) | Disabled by default | Enabled by default | Yes | Disabled by default​[[149]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-2008R1TLS1.2-194) (KB4019276) | Disabled by default​[[149]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-2008R1TLS1.2-194) (KB4019276) | No | Yes | Yes | Yes[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Mitigated | Not affected | Vulnerable | Vulnerable | Mitigated [[147]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-031-192) | Mitigated [[148]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-055-193) | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| [8](https://en.wikipedia.org/wiki/Internet_Explorer_8), [9](https://en.wikipedia.org/wiki/Internet_Explorer_9), [10](https://en.wikipedia.org/wiki/Internet_Explorer_10) | | [Windows 7](https://en.wikipedia.org/wiki/Windows_7) [Server 2008 R2](https://en.wikipedia.org/wiki/Windows_Server_2008_R2) | Disabled by default | Enabled by default | Yes | Disabled by default [[150]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Win7TLS1.1-195) | Disabled by default [[150]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Win7TLS1.1-195) | No | Yes | Yes | Yes | Mitigated | Not affected | Vulnerable | Lowest priority [[151]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-20141111msblog-196)[[n 32]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4fallbackWin7-198) | Mitigated [[147]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-031-192) | Mitigated [[148]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-055-193) | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| [10](https://en.wikipedia.org/wiki/Internet_Explorer_10) | | [Windows 8](https://en.wikipedia.org/wiki/Windows_8) | Disabled by default | Enabled by default | Yes | Disabled by default [[150]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Win7TLS1.1-195) | Disabled by default [[150]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Win7TLS1.1-195) | No | Yes | Yes | Yes | Mitigated | Not affected | Vulnerable | Lowest priority [[151]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-20141111msblog-196)[[n 32]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4fallbackWin7-198) | Mitigated [[147]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-031-192) | Mitigated [[148]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-055-193) | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| [Server 2012](https://en.wikipedia.org/wiki/Windows_Server_2012) |
| [**Microsoft Edge**](https://en.wikipedia.org/wiki/Microsoft_Edge) **(79+)** [**(Chromium based)**](https://en.wikipedia.org/wiki/Chromium_(web_browser)) **OS independent** [**Internet Explorer 11**](https://en.wikipedia.org/wiki/Internet_Explorer_11)[**[n 28]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-d-183) | [11](https://en.wikipedia.org/wiki/Internet_Explorer_11) | | [Windows 7](https://en.wikipedia.org/wiki/Windows_7) [Server 2008 R2](https://en.wikipedia.org/wiki/Windows_Server_2008_R2) | Disabled by default | Disabled by default [[n 33]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ie11_ssl3-199) | Yes | Yes[[153]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-IE11_TLS_1.1_1.2-200) | Yes[[153]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-IE11_TLS_1.1_1.2-200) | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated [[n 33]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ie11_ssl3-199) | Disabled by default​[[157]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-204) | Mitigated [[147]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-031-192) | Mitigated [[148]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-055-193) | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| [11](https://en.wikipedia.org/wiki/Internet_Explorer_11)[[158]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-205) | | [Windows 8.1](https://en.wikipedia.org/wiki/Windows_8.1) | Disabled by default | Disabled by default [[n 33]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ie11_ssl3-199) | Yes | Yes[[153]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-IE11_TLS_1.1_1.2-200) | Yes[[153]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-IE11_TLS_1.1_1.2-200) | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated [[n 33]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ie11_ssl3-199) | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated [[147]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-031-192) | Mitigated [[148]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MS15-055-193) | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| [Server 2012](https://en.wikipedia.org/wiki/Windows_Server_2012) [Server 2012 R2](https://en.wikipedia.org/wiki/Windows_Server_2012_R2) |
| **Browser** | **Version** | | **Platforms** | **SSL 2.0 (insecure)** | **SSL 3.0 (insecure)** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** | **EV certificate** | **SHA-2 certificate** | **ECDSA certificate** | **BEAST** | **CRIME** | **POODLE (SSLv3)** | **RC4** | **FREAK** | **Logjam** | **Protocol selection by user** |
| [**Microsoft Edge**](https://en.wikipedia.org/wiki/Microsoft_Edge) **(79+)** [**(Chromium based)**](https://en.wikipedia.org/wiki/Chromium_(web_browser)) **OS independent** [**Microsoft Edge**](https://en.wikipedia.org/wiki/Microsoft_Edge) **(12–18)** [**(EdgeHTML based)**](https://en.wikipedia.org/wiki/EdgeHTML) **Client only** [**Internet Explorer 11**](https://en.wikipedia.org/wiki/Internet_Explorer_11)[**[n 28]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-d-183) | 11 | 12–13 | [Windows 10](https://en.wikipedia.org/wiki/Windows_10) v1507–v1511 | Disabled by default | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11 | 14–18 | [Windows 10](https://en.wikipedia.org/wiki/Windows_10) v1607–v1803 | No[[159]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Windows10_v1607schannel-206) | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11 | | [Windows Server (SAC)](https://en.wikipedia.org/wiki/Windows_Server_2016) v1709–v1803 | No | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11 | 18 | [Windows 10](https://en.wikipedia.org/wiki/Windows_10) v1809 | No | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11 | | [Windows Server (SAC)](https://en.wikipedia.org/wiki/Windows_Server_2019) v1809 | No | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11 | 18 | [Windows 10](https://en.wikipedia.org/wiki/Windows_10) v1903 | No | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11 | | [Windows Server (SAC)](https://en.wikipedia.org/wiki/Windows_Server_2019) v1903 | No | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11 | 18 | [Windows 10](https://en.wikipedia.org/wiki/Windows_10) v1909 | No | Disabled by default | Yes | Yes | Yes | Disabled by default (experimental)[[160]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-207) | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11 | | [Windows Server (SAC)](https://en.wikipedia.org/wiki/Windows_Server_2019) v1909 | No | Disabled by default | Yes | Yes | Yes | Disabled by default (experimental) | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11 | 18 | [Windows 10](https://en.wikipedia.org/wiki/Windows_10) v2004 | No | Disabled by default | Yes | Yes | Yes | Disabled by default (experimental) | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11 | | [Windows Server (SAC)](https://en.wikipedia.org/wiki/Windows_Server_2019) v2004 | No | Disabled by default | Yes | Yes | Yes | Disabled by default (experimental) | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| [**Microsoft Edge**](https://en.wikipedia.org/wiki/Microsoft_Edge) **(79+)** [**(Chromium based)**](https://en.wikipedia.org/wiki/Chromium_(web_browser)) **OS independent** [**Internet Explorer 11**](https://en.wikipedia.org/wiki/Internet_Explorer_11)[**[n 28]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-d-183) | 11 | | [Windows 10](https://en.wikipedia.org/wiki/Windows_10) 20H2 | No | Disabled by default | Yes | Yes | Yes | Disabled by default (experimental) | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11 | | [Windows Server (SAC)](https://en.wikipedia.org/wiki/Windows_Server_2019) 20H2 | No | Disabled by default | Yes | Yes | Yes | Disabled by default (experimental) | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| [**Microsoft Edge**](https://en.wikipedia.org/wiki/Microsoft_Edge) **(79+)** [**(Chromium based)**](https://en.wikipedia.org/wiki/Chromium_(web_browser)) **OS independent** [**Internet Explorer 11**](https://en.wikipedia.org/wiki/Internet_Explorer_11)[**[n 28]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-d-183) | 11 | | [Windows 10](https://en.wikipedia.org/wiki/Windows_10) LTSB 2015 (v1507) | Disabled by default | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11 | | [Windows 10](https://en.wikipedia.org/wiki/Windows_10) LTSB 2016 (v1607) | No[[159]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Windows10_v1607schannel-206) | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11 | | [Windows Server 2016](https://en.wikipedia.org/wiki/Windows_Server_2016) v1607 (LTSB) | No[[159]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Windows10_v1607schannel-206) | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11 | | [Windows 10](https://en.wikipedia.org/wiki/Windows_10) LTSC 2019 (v1809) | No | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| 11 | | [Windows Server 2019](https://en.wikipedia.org/wiki/Windows_Server_2019) v1809 (LTSC) | No | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | Yes[[n 10]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viasetting-103) |
| **Browser** | **Version** | | **Platforms** | **SSL 2.0 (insecure)** | **SSL 3.0 (insecure)** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** | **EV certificate** | **SHA-2 certificate** | **ECDSA certificate** | **BEAST** | **CRIME** | **POODLE (SSLv3)** | **RC4** | **FREAK** | **Logjam** | **Protocol selection by user** |
| [**Microsoft Internet Explorer Mobile**](https://en.wikipedia.org/wiki/Internet_Explorer_Mobile)[**[n 28]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-d-183) | [7, 9](https://en.wikipedia.org/wiki/Internet_Explorer_Mobile) | | [Windows Phone 7, 7.5, 7.8](https://en.wikipedia.org/wiki/Windows_Phone_7) | Disabled by default [[144]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-IE7HTTPS-189) | Enabled by default | Yes | No [[*citation needed*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)] | No [[*citation needed*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)] | No | No [[*citation needed*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)] | Yes | Yes[[161]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC2-208) | Unknown | Not affected | Vulnerable | Vulnerable | Vulnerable | Vulnerable | Only with 3rd party tools[[n 34]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viaregistry-210) |
| [10](https://en.wikipedia.org/wiki/Internet_Explorer_Mobile) | | [Windows Phone 8](https://en.wikipedia.org/wiki/Windows_Phone_8) | Disabled by default | Enabled by default | Yes | Disabled by default [[163]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-IEMobile10-211) | Disabled by default [[163]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-IEMobile10-211) | No | No [[*citation needed*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)] | Yes | Yes[[164]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ssllabs.com-212) | Mitigated | Not affected | Vulnerable | Vulnerable | Vulnerable | Vulnerable | Only with 3rd party tools[[n 34]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viaregistry-210) |
| [11](https://en.wikipedia.org/wiki/Internet_Explorer_Mobile) | | [Windows Phone 8.1](https://en.wikipedia.org/wiki/Windows_Phone_8.1) | Disabled by default | Enabled by default | Yes | Yes[[165]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-IEMobile11-213) | Yes[[165]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-IEMobile11-213) | No | No [[*citation needed*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)] | Yes | Yes | Mitigated | Not affected | Vulnerable | Only as fallback [[n 15]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4fallback-120)[[166]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Win8.1RC4-214)[[167]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Win8.1RC4Qualys-215) | Vulnerable | Vulnerable | Only with 3rd party tools[[n 34]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-viaregistry-210) |
| [**Microsoft Edge**](https://en.wikipedia.org/wiki/Microsoft_Edge) **(13–15)** [**(EdgeHTML based)**](https://en.wikipedia.org/wiki/EdgeHTML)[**[n 35]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Edge_engine-216) | [13](https://en.wikipedia.org/wiki/Microsoft_Edge) | | [Windows 10 Mobile](https://en.wikipedia.org/wiki/Windows_10_Mobile) v1511 | Disabled by default | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | No |
| [14, 15](https://en.wikipedia.org/wiki/Microsoft_Edge) | | [Windows 10 Mobile](https://en.wikipedia.org/wiki/Windows_10_Mobile) v1607–v1709 | No[[159]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Windows10_v1607schannel-206) | Disabled by default | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | No |
| **Browser** | **Version** | | **Platforms** | **SSL 2.0 (insecure)** | **SSL 3.0 (insecure)** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** | **EV certificate** | **SHA-2 certificate** | **ECDSA certificate** | **BEAST** | **CRIME** | **POODLE (SSLv3)** | **RC4** | **FREAK** | **Logjam** | **Protocol selection by user** |
| [**Apple Safari**](https://en.wikipedia.org/wiki/Safari_(web_browser))[**[n 36]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-g-221) | 1 | | Mac OS X [10.2](https://en.wikipedia.org/wiki/Mac_OS_X_v10.2), [10.3](https://en.wikipedia.org/wiki/Mac_OS_X_Panther) | No[[172]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SafariSSL2-222) | Yes | Yes | No | No | No | No | No | No | Vulnerable | Not affected | Vulnerable | Vulnerable | Vulnerable | Vulnerable | No |
| 2–5 | | Mac OS X [10.4](https://en.wikipedia.org/wiki/Mac_OS_X_Tiger), [10.5](https://en.wikipedia.org/wiki/Mac_OS_X_Leopard), [Win XP](https://en.wikipedia.org/wiki/Windows_XP) | No | Yes | Yes | No | No | No | since v3.2 | No | No | Vulnerable | Not affected | Vulnerable | Vulnerable | Vulnerable | Vulnerable | No |
| 3–5 | | [Vista](https://en.wikipedia.org/wiki/Windows_Vista), [Win 7](https://en.wikipedia.org/wiki/Windows_7) | No | Yes | Yes | No | No | No | since v3.2 | No | Yes[[161]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC2-208) | Vulnerable | Not affected | Vulnerable | Vulnerable | Vulnerable | Vulnerable | No |
| 4–6 | | Mac OS X [10.6](https://en.wikipedia.org/wiki/Mac_OS_X_Snow_Leopard), [10.7](https://en.wikipedia.org/wiki/Mac_OS_X_Lion) | No | Yes | Yes | No | No | No | Yes | Yes[[71]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SHA-2-88) | Yes[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Vulnerable | Not affected | Vulnerable | Vulnerable | Vulnerable | Vulnerable | No |
| 6 | | [OS X 10.8](https://en.wikipedia.org/wiki/OS_X_Mountain_Lion) | No | Yes | Yes | No | No | No | Yes | Yes | Yes[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Mitigated [[n 37]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-i-226) | Not affected | Mitigated [[n 38]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-safari-poodle-229) | Vulnerable [[n 38]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-safari-poodle-229) | Mitigated [[178]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-freak-OSX-230) | Vulnerable | No |
| 7, 9 | | [OS X 10.9](https://en.wikipedia.org/wiki/OS_X_Mavericks) | No | Yes | Yes | Yes[[179]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MavericksTLS1.2-231) | Yes[[179]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MavericksTLS1.2-231) | No | Yes | Yes | Yes | Mitigated [[174]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-beast-mitigations-mavericks-224) | Not affected | Mitigated [[n 38]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-safari-poodle-229) | Vulnerable [[n 38]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-safari-poodle-229) | Mitigated [[178]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-freak-OSX-230) | Vulnerable | No |
| 8–10 | | [OS X 10.10](https://en.wikipedia.org/wiki/OS_X_Yosemite) | No | Yes | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated [[n 38]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-safari-poodle-229) | Lowest priority [[180]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ssltest-safari-OSX1010-232)[[n 38]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-safari-poodle-229) | Mitigated [[178]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-freak-OSX-230) | Mitigated [[181]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-logjam-OSX-233) | No |
| 9–11 | | [OS X 10.11](https://en.wikipedia.org/wiki/OS_X_El_Capitan) | No | No | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Not affected | Lowest priority | Mitigated | Mitigated | No |
| 10-12 | | [macOS 10.12](https://en.wikipedia.org/wiki/MacOS_Sierra) | No | No | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | No |
| 11, 12 | 13 | [macOS 10.13](https://en.wikipedia.org/wiki/MacOS_High_Sierra) | No | No | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | No |
| 12 | 13 | [macOS 10.14](https://en.wikipedia.org/wiki/MacOS_Mojave) | No | No | Yes | Yes | Yes | Yes (since macOS 10.14.4)[[182]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Pauly-234) | Yes | Yes | Yes | Mitigated | Not affected | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | No |
| 13 | | [macOS 10.15](https://en.wikipedia.org/wiki/MacOS_Catalina) | No | No | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Mitigated | Not affected | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | No |
| **Browser** | **Version** | | **Platforms** | **SSL 2.0 (insecure)** | **SSL 3.0 (insecure)** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** | **EV certificate** | **SHA-2 certificate** | **ECDSA certificate** | **BEAST** | **CRIME** | **POODLE (SSLv3)** | **RC4** | **FREAK** | **Logjam** | **Protocol selection by user** |
| [**Apple Safari**](https://en.wikipedia.org/wiki/Safari_(web_browser)) **(mobile)** [**[n 39]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-h-238) | 3 | | iPhone OS [1](https://en.wikipedia.org/wiki/IPhone_OS_1), [2](https://en.wikipedia.org/wiki/IPhone_OS_2) | No[[186]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-239) | Yes | Yes | No | No | No | No | No | No | Vulnerable | Not affected | Vulnerable | Vulnerable | Vulnerable | Vulnerable | No |
| 4, 5 | | [iPhone OS 3](https://en.wikipedia.org/wiki/IPhone_OS_3), [iOS 4](https://en.wikipedia.org/wiki/IOS_4) | No | Yes | Yes | No | No | No | Yes[[187]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-theiphoneblog.com-240) | Yes | since iOS 4[[161]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC2-208) | Vulnerable | Not affected | Vulnerable | Vulnerable | Vulnerable | Vulnerable | No |
| 5, 6 | | iOS [5](https://en.wikipedia.org/wiki/IOS_5), [6](https://en.wikipedia.org/wiki/IOS_6) | No | Yes | Yes | Yes[[183]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-iOS5TLS1.2-235) | Yes[[183]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-iOS5TLS1.2-235) | No | Yes | Yes | Yes | Vulnerable | Not affected | Vulnerable | Vulnerable | Vulnerable | Vulnerable | No |
| 7 | | [iOS 7](https://en.wikipedia.org/wiki/IOS_7) | No | Yes | Yes | Yes | Yes | No | Yes | Yes | Yes[[188]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ReferenceC-241) | Mitigated [[189]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-beast-mitigations-ios7-242) | Not affected | Vulnerable | Vulnerable | Vulnerable | Vulnerable | No |
| 8 | | [iOS 8](https://en.wikipedia.org/wiki/IOS_8) | No | Yes | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Mitigated [[n 38]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-safari-poodle-229) | Lowest priority [[190]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ssltest-safari-iOS812-243)[[n 38]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-safari-poodle-229) | Mitigated [[191]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-freak-iOS82-244) | Mitigated [[192]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-logjam-iOS84-245) | No |
| 9 | | [iOS 9](https://en.wikipedia.org/wiki/IOS_9) | No | No | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Not affected | Lowest priority | Mitigated | Mitigated | No |
| 10–11 | | iOS [10](https://en.wikipedia.org/wiki/IOS_10), [11](https://en.wikipedia.org/wiki/IOS_11) | No | No | Yes | Yes | Yes | No | Yes | Yes | Yes | Mitigated | Not affected | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | No |
| 12 | | [iOS 12](https://en.wikipedia.org/wiki/IOS_12) | No | No | Yes | Yes | Yes | Yes (since iOS 12.2)[[182]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Pauly-234) | Yes | Yes | Yes | Mitigated | Not affected | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | No |
| 13 | | [iOS 13](https://en.wikipedia.org/wiki/IOS_13) | No | No | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Mitigated | Not affected | Not affected | Disabled by default​[[n 16]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4-disabled-124) | Mitigated | Mitigated | No |
| [iPadOS 13](https://en.wikipedia.org/wiki/IPadOS) |
| **Browser** | **Version** | | **Platforms** | **SSL 2.0 (insecure)** | **SSL 3.0 (insecure)** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** | **EV** [**[n 3]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-evssl-86) | **SHA-2** | **ECDSA** | **BEAST**[**[n 4]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-BEASTmitigation-90) | **CRIME**[**[n 5]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-CRIMEmitigation-91) | **POODLE (SSLv3)**[**[n 6]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-POODLEmitigation-92) | **RC4**[**[n 7]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-RC4mitigation-93) | **FREAK**[**[73]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-FREAK-1-94)[**[74]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-note-FREAK-2-95) | **Logjam** | **Protocol selection by user** |
| **SSL protocols** | | **TLS protocols** | | | | **Certificate Support** | | | **Vulnerabilities fixed** | | | | | |
| [**Google Android OS**](https://en.wikipedia.org/wiki/Android_(OS))[**[193]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-android-tls-246) | Android [1.0–4.0.4](https://en.wikipedia.org/wiki/Android_version_history) | | | No | Enabled by default | Yes | No | No | No | Unknown | Yes[[71]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-SHA-2-88) | since 3.0[[161]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC2-208)[[72]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ECC-89) | Unknown | Unknown | Vulnerable | Vulnerable | Vulnerable | Vulnerable | No |
| Android [4.1–4.4.4](https://en.wikipedia.org/wiki/Android_version_history) | | | No | Enabled by default | Yes | Disabled by default​[[194]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Android-SSLSocket-247) | Disabled by default​[[194]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Android-SSLSocket-247) | No | Unknown | Yes | Yes | Unknown | Unknown | Vulnerable | Vulnerable | Vulnerable | Vulnerable | No |
| Android [5.0–5.0.2](https://en.wikipedia.org/wiki/Android_Lollipop) | | | No | Enabled by default | Yes | Yes[[194]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Android-SSLSocket-247)[[195]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-android-5-tls-248) | Yes[[194]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Android-SSLSocket-247)[[195]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-android-5-tls-248) | No | Unknown | Yes | Yes | Unknown | Unknown | Vulnerable | Vulnerable | Vulnerable | Vulnerable | No |
| Android [5.1–5.1.1](https://en.wikipedia.org/wiki/Android_Lollipop) | | | No | Disabled by default [[*citation needed*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)] | Yes | Yes | Yes | No | Unknown | Yes | Yes | Unknown | Unknown | Not affected | Only as fallback [[n 15]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RC4fallback-120) | Mitigated | Mitigated | No |
| Android [6.0](https://en.wikipedia.org/wiki/Android_Marshmallow)–[7.1.2](https://en.wikipedia.org/wiki/Android_Nougat) | | | No | Disabled by default [[*citation needed*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)] | Yes | Yes | Yes | No | Unknown | Yes | Yes | Unknown | Unknown | Not affected | Disabled by default | Mitigated | Mitigated | No |
| Android [8.0](https://en.wikipedia.org/wiki/Android_Oreo)–[9.0](https://en.wikipedia.org/wiki/Android_Pie) | | | No | No [[196]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Android_8.0_Behavior_Changes-249) | Yes | Yes | Yes | No | Unknown | Yes | Yes | Unknown | Unknown | Not affected | Disabled by default | Mitigated | Mitigated | No |
| Android [10.0](https://en.wikipedia.org/wiki/Android_10) | | | No | No | Yes | Yes | Yes | Yes | Unknown | Yes | Yes | Unknown | Unknown | Not affected | Disabled by default | Mitigated | Mitigated | No |
| Android [11.0](https://en.wikipedia.org/wiki/Android_11) | | | No | No | Yes | Yes | Yes | Yes | Unknown | Yes | Yes | Unknown | Unknown | Not affected | Disabled by default | Mitigated | Mitigated | No |
| **Browser** | **Version** | | **Platforms** | **SSL 2.0 (insecure)** | **SSL 3.0 (insecure)** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** | **EV certificate** | **SHA-2 certificate** | **ECDSA certificate** | **BEAST** | **CRIME** | **POODLE (SSLv3)** | **RC4** | **FREAK** | **Logjam** | **Protocol selection by user** |

|  |  |  |
| --- | --- | --- |
| **Color or Note** | | **Significance** |
| **Browser version** | **Platform** |
| Browser version | Operating system | Future release; under development |
| Browser version | Operating system | Current latest release |
| Browser version | Operating system | Former release; still supported |
| Browser version | Operating system | Former release; long-term support still active, but will end in less than 12 months |
| Browser version | Operating system | Former release; no longer supported |
| n/a | Operating system | Mixed / Unspecified |
| Operating system (Version+) | Minimum required operating system version (for supported versions of the browser) |
| ~~Operating system~~ | No longer supported for this operating system |

Notes

 Does the browser have mitigations or is not vulnerable for the known attacks. Note actual security depends on other factors such as negotiated cipher, encryption strength etc (see [§ Cipher](https://en.wikipedia.org/wiki/Transport_Layer_Security#Cipher) table).

  Whether a user or administrator can choose the protocols to be used or not. If yes, several attacks such as BEAST (vulnerable in SSL 3.0 and TLS 1.0) or POODLE (vulnerable in SSL 3.0) can be avoided.

  Whether EV SSL and DV SSL (normal SSL) can be distinguished by indicators (green lock icon, green address bar, etc.) or not.

  e.g. 1/n-1 record splitting.

  e.g. Disabling header compression in HTTPS/SPDY.

*   Complete mitigations; disabling SSL 3.0 itself, "anti-POODLE record splitting". "Anti-POODLE record splitting" is effective only with client-side implementation and valid according to the SSL 3.0 specification, however, it may also cause compatibility issues due to problems in server-side implementations.
* Partial mitigations; disabling fallback to SSL 3.0, TLS\_FALLBACK\_SCSV, disabling cipher suites with [CBC mode of operation](https://en.wikipedia.org/wiki/Block_cipher_mode_of_operation#Cipher-block_chaining_(CBC)). If the server also supports TLS\_FALLBACK\_SCSV, the POODLE attack will fail against this combination of server and browser, but connections where the server does not support TLS\_FALLBACK\_SCSV and does support SSL 3.0 will still be vulnerable. If disabling cipher suites with CBC mode of operation in SSL 3.0, only cipher suites with RC4 are available, RC4 attacks become easier.
* When disabling SSL 3.0 manually, POODLE attack will fail.
*   Complete mitigation; disabling cipher suites with RC4.
* Partial mitigations to keeping compatibility with old systems; setting the priority of RC4 to lower.

  [Google Chrome](https://en.wikipedia.org/wiki/Google_Chrome) (and [Chromium](https://en.wikipedia.org/wiki/Chromium_(web_browser))) supports TLS 1.0, and TLS 1.1 from version 22 (it was added, then dropped from version 21). TLS 1.2 support has been added, then dropped from Chrome 29.[[75]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Google-96)[[76]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-googlechromereleases.blogspot.co.uk-97)[[77]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Chromium_Project-98)

  Uses the TLS implementation provided by [BoringSSL](https://en.wikipedia.org/wiki/BoringSSL) for Android, OS X, and Windows[[78]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-The_Chromium_Project:_BoringSSL-100) or by [NSS](https://en.wikipedia.org/wiki/Network_Security_Services) for Linux. Google is switching the TLS library used in Chrome to BoringSSL from NSS completely.

  configure enabling/disabling of each protocols via setting/option (menu name is dependent on browsers)

  configure the maximum and the minimum version of enabling protocols with command-line option

  TLS\_FALLBACK\_SCSV is implemented.[[86]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chrome-scsv-111) Fallback to SSL 3.0 is disabled since version 39.[[87]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chrome-poodle-112)

  In addition to TLS\_FALLBACK\_SCSV and disabling a fallback to SSL 3.0, SSL 3.0 itself is disabled by default.[[87]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chrome-poodle-112)

  configure the minimum version of enabling protocols via chrome://flags[[91]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chromium693963003-117) (the maximum version can be configured with command-line option)

  Only when no cipher suites with other than RC4 is available, cipher suites with RC4 will be used as a fallback.

  All RC4 cipher suites are disabled by default.

  Uses the TLS implementation provided by [NSS](https://en.wikipedia.org/wiki/Network_Security_Services). As of Firefox 22, Firefox supports only TLS 1.0 despite the bundled NSS supporting TLS 1.1. Since Firefox 23, TLS 1.1 can be enabled, but was not enabled by default due to issues. Firefox 24 has TLS 1.2 support disabled by default. TLS 1.1 and TLS 1.2 have been enabled by default in Firefox 27 release.

  configure the maximum and the minimum version of enabling protocols via about:config

  SSL 3.0 itself is disabled by default.[[108]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-firefox-poodle-140) In addition, fallback to SSL 3.0 is disabled since version 34,[[110]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-firefox-fallback-disabled-142) and TLS\_FALLBACK\_SCSV is implemented since 35.0 and ESR 31.3.[[108]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-firefox-poodle-140)[[111]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-bugzilla.mozilla.org-143)

  [Opera](https://en.wikipedia.org/wiki/Opera_(web_browser)) 10 added support for TLS 1.2 as of [Presto](https://en.wikipedia.org/wiki/Presto_(layout_engine)) 2.2. Previous support was for TLS 1.0 and 1.1. TLS 1.1 and 1.2 are disabled by default (except for version 9[[121]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Opera_9.0_for_Windows_Changelog-154) that enabled TLS 1.1 by default).

  SSL 3.0 is disabled by default remotely since October 15, 2014[[130]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-OperaPOODLE-165)

  TLS support of Opera 14 and above is same as that of Chrome, because Opera has migrated to [Chromium](https://en.wikipedia.org/wiki/Chromium_(web_browser)) backend (Opera 14 for Android is based on Chromium 26 with [WebKit](https://en.wikipedia.org/wiki/WebKit),[[135]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-dev.opera.com-170) and Opera 15 and above are based on Chromium 28 and above with [Blink](https://en.wikipedia.org/wiki/Blink_(layout_engine))[[136]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ReferenceB-171)).

  TLS\_FALLBACK\_SCSV is implemented.[[139]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-opera20-175)

  SSL 3.0 is enabled by default, with some mitigations against known vulnerabilities such as BEAST and POODLE implemented.[[130]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-OperaPOODLE-165)

  In addition to TLS\_FALLBACK\_SCSV, "anti-POODLE record splitting" is implemented.[[130]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-OperaPOODLE-165)

  In addition to TLS\_FALLBACK\_SCSV and "anti-POODLE record splitting", SSL 3.0 itself is disabled by default.[[91]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chromium693963003-117)

  configure the minimum version of enabling protocols via opera://flags[[91]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-chromium693963003-117) (the maximum version can be configured with command-line option)

  IE uses the TLS implementation of the Microsoft Windows operating system provided by the [SChannel](https://en.wikipedia.org/wiki/SChannel) security support provider. TLS 1.1 and 1.2 are disabled by default until IE11.[[140]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Microsoft-181)[[141]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-msdn.microsoft.com-182) Cite error: The named reference "note-d" was defined multiple times with different content (see the [help page](https://en.wikipedia.org/wiki/Help:Cite_errors/Cite_error_references_duplicate_key)).

  [Windows NT 3.1](https://en.wikipedia.org/wiki/Windows_NT_3.1) supports IE 1–2, [Windows NT 3.5](https://en.wikipedia.org/wiki/Windows_NT_3.5) supports IE 1–3, [Windows NT 3.51](https://en.wikipedia.org/wiki/Windows_NT_3.51) and [Windows NT 4.0](https://en.wikipedia.org/wiki/Windows_NT_4.0) supports IE 1–6

  Windows XP as well as Server 2003 and older support only weak ciphers like 3DES and RC4 out of the box.[[145]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-TLS_Cipher_Suites-190) The weak ciphers of these SChannel version are not only used for IE, but also for other Microsoft products running on this OS, like Office or Windows Update. Only Windows Server 2003 can get a manually update to support AES ciphers by KB948963[[146]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Archived_copy-191)

  MS13-095 or MS14-049 for 2003 and XP-64 or SP3 for XP (32-bit)

  RC4 can be disabled except as a fallback (Only when no cipher suites with other than RC4 is available, cipher suites with RC4 will be used as a fallback.)[[152]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-win7-8rc4-197)

  Fallback to SSL 3.0 is sites blocked by default in Internet Explorer 11 for Protected Mode.[[154]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-201)[[155]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-support.microsoft.com-202) SSL 3.0 is disabled by default in Internet Explorer 11 since April 2015.[[156]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-technet.microsoft.com-203)

  Could be disabled via registry editing but need 3rd Party tools to do this.[[162]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-XDA-209)

  Cite error: The named reference Edge engine was invoked but never defined (see the [help page](https://en.wikipedia.org/wiki/Help:Cite_errors/Cite_error_references_no_text)).

  Safari uses the operating system implementation on Mac OS X, Windows (XP, Vista, 7)[[168]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Adrian-217) with unknown version,[[169]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Apple-218) Safari 5 is the last version available for Windows. OS X 10.8 on have SecureTransport support for TLS 1.1 and 1.2[[170]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-curl.haxx.se-219) Qualys SSL report simulates Safari 5.1.9 connecting with TLS 1.0 not 1.1 or 1.2[[171]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Qualys_SSL_Report:_google.co.uk-220)

  In September 2013, Apple implemented [BEAST](https://en.wikipedia.org/wiki/Transport_Layer_Security#BEAST) mitigation in [OS X 10.8](https://en.wikipedia.org/wiki/OS_X_10.8) (Mountain Lion), but it was not turned on by default, resulting in Safari still being theoretically vulnerable to the BEAST attack on that platform.[[173]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-BEAST-Still-223)[[174]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-beast-mitigations-mavericks-224) BEAST mitigation has been enabled by default from OS X 10.8.5 updated in February 2014.[[175]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-beast-mitigations-10.8-225)

  Because Apple removed support for all CBC protocols in SSL 3.0 to mitigate POODLE,[[176]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-227)[[177]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-support.apple.com-228) this leaves only RC4, which is also completely broken by the RC4 attacks in SSL 3.0.

* 1.  Mobile Safari and third-party software utilizing the system UIWebView library use the [iOS](https://en.wikipedia.org/wiki/IOS) operating system implementation, which supports TLS 1.2 as of iOS 5.0.[[183]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-iOS5TLS1.2-235)[[184]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Liebowitz-236)[[185]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-MWR_Info_Security-237)

**Libraries**

Main article: [Comparison of TLS implementations](https://en.wikipedia.org/wiki/Comparison_of_TLS_implementations)

Most SSL and TLS programming libraries are [free and open source software](https://en.wikipedia.org/wiki/Free_and_open_source_software).

* [BoringSSL](https://en.wikipedia.org/wiki/OpenSSL#BoringSSL), a fork of OpenSSL for Chrome/Chromium and Android as well as other Google applications.
* [Botan](https://en.wikipedia.org/wiki/Botan_(programming_library)), a BSD-licensed cryptographic library written in C++.
* [cryptlib](https://en.wikipedia.org/wiki/Cryptlib): a portable open source cryptography library (includes TLS/SSL implementation)
* [Delphi](https://en.wikipedia.org/wiki/Delphi_(programming_language)) programmers may use a library called [Indy](https://en.wikipedia.org/wiki/Internet_Direct) which utilizes [OpenSSL](https://en.wikipedia.org/wiki/OpenSSL) or alternatively ICS which supports TLS 1.3 now.
* [GnuTLS](https://en.wikipedia.org/wiki/GnuTLS): a free implementation (LGPL licensed)
* [Java Secure Socket Extension](https://en.wikipedia.org/wiki/Java_Secure_Socket_Extension): a [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) implementation included in the [Java Runtime Environment](https://en.wikipedia.org/wiki/Java_Runtime_Environment) supported TLS 1.1 and 1.2 starting with Java 7. (TLS 1.1/1.2 were initially disabled by default for client on Java 7, but were enabled in January 2017.[[197]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Oracle-250)) Java 11 supports TLS 1.3.[[198]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-bugs.openjdk.java.net-251)
* [LibreSSL](https://en.wikipedia.org/wiki/LibreSSL): a fork of OpenSSL by OpenBSD project.
* [MatrixSSL](https://en.wikipedia.org/wiki/MatrixSSL): a dual licensed implementation
* [mbed TLS](https://en.wikipedia.org/wiki/Mbed_TLS) (previously PolarSSL): A tiny SSL library implementation for embedded devices that is designed for ease of use
* [Network Security Services](https://en.wikipedia.org/wiki/Network_Security_Services): [FIPS 140](https://en.wikipedia.org/wiki/FIPS_140) validated open source library
* [OpenSSL](https://en.wikipedia.org/wiki/OpenSSL): a free implementation (BSD license with some extensions)
* [RSA BSAFE](https://en.wikipedia.org/wiki/RSA_BSAFE) Micro Edition Suite: a multi-platform implementation of TLS written in [C](https://en.wikipedia.org/wiki/C_(programming_language)) using a FIPS-validated cryptographic module
* [RSA BSAFE](https://en.wikipedia.org/wiki/RSA_BSAFE) SSL-J: a TLS library providing both a proprietary API and [JSSE](https://en.wikipedia.org/wiki/Java_Secure_Socket_Extension) API, using FIPS-validated cryptographic module
* [SChannel](https://en.wikipedia.org/wiki/Security_Support_Provider_Interface): an implementation of SSL and TLS [Microsoft Windows](https://en.wikipedia.org/wiki/Microsoft_Windows) as part of its package.
* [Secure Transport](https://en.wikipedia.org/wiki/Secure_Transport): an implementation of SSL and TLS used in [OS X](https://en.wikipedia.org/wiki/OS_X) and [iOS](https://en.wikipedia.org/wiki/IOS) as part of their packages.
* [wolfSSL](https://en.wikipedia.org/wiki/WolfSSL) (previously CyaSSL): Embedded SSL/TLS Library with a strong focus on speed and size.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Library support for TLS/SSL | | | | | | |
| **Implementation** | **SSL 2.0 (insecure)** | **SSL 3.0 (insecure)** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** |
| [**Botan**](https://en.wikipedia.org/wiki/Botan_(programming_library)) | No | No[[199]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Botan_1.11.13-252) | Yes | Yes | Yes |  |
| [**cryptlib**](https://en.wikipedia.org/wiki/Cryptlib) | No | Disabled by default at compile time | Yes | Yes | Yes |  |
| [**GnuTLS**](https://en.wikipedia.org/wiki/GnuTLS) | No[[a]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cnote_a_grp_protocollibrary-table) | Disabled by default[[200]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-GnuTLS-3.4.0-253) | Yes | Yes | Yes | Yes[[201]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-GnuTLS-3.6.4-254) |
| [**Java Secure Socket Extension**](https://en.wikipedia.org/wiki/Java_Secure_Socket_Extension) | No[[a]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cnote_a_grp_protocollibrary-table) | Disabled by default[[202]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-oracle.com-255) | Yes | Yes | Yes | Yes |
| [**LibreSSL**](https://en.wikipedia.org/wiki/LibreSSL) | No[[203]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-OpenBSD5.6-256) | No[[204]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-libressl-2.3-257) | Yes | Yes | Yes |  |
| [**MatrixSSL**](https://en.wikipedia.org/wiki/MatrixSSL) | No | Disabled by default at compile time[[205]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Matrix-POODLE-258) | Yes | Yes | Yes | yes (draft version) |
| [**mbed TLS**](https://en.wikipedia.org/wiki/Mbed_TLS) **(previously PolarSSL)** | No | Disabled by default[[206]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-mbed-2.0-259) | Yes | Yes | Yes |  |
| [**Network Security Services**](https://en.wikipedia.org/wiki/Network_Security_Services) | No[[b]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cnote_b_grp_protocolsupport) | Disabled by default[[207]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-NSS-3.19-260) | Yes | Yes[[208]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-NSS-3.14-261) | Yes[[209]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-NSS-3.15.1-262) | Yes[[210]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-NSS_3.39_release_notes-263) |
| [**OpenSSL**](https://en.wikipedia.org/wiki/OpenSSL) | No[[211]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-OpenSSL_1.1.0_Series_Release_Notes-264) | Enabled by default | Yes | Yes[[212]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-openssl-1.0.1-note-265) | Yes[[212]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-openssl-1.0.1-note-265) | Yes[[213]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-OpenSSL_1.1.1_Is_Released-266) |
| [**RSA BSAFE**](https://en.wikipedia.org/wiki/RSA_BSAFE) **Micro Edition Suite** | No | Disabled by default | Yes | Yes | Yes | Not yet |
| [**RSA BSAFE**](https://en.wikipedia.org/wiki/RSA_BSAFE) **SSL-J** | No | Disabled by default | Yes | Yes | Yes | Not yet |
| [**SChannel XP / 2003**](https://en.wikipedia.org/wiki/SChannel)[**[214]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-XP2003schannel-267) | Disabled by default by MSIE 7 | Enabled by default | Enabled by default by MSIE 7 | No | No | No |
| [**SChannel Vista**](https://en.wikipedia.org/wiki/SChannel)[**[215]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Vista2008schannel-268) | Disabled by default | Enabled by default | Yes | No | No | No |
| [**SChannel 2008**](https://en.wikipedia.org/wiki/SChannel)[**[215]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Vista2008schannel-268) | Disabled by default | Enabled by default | Yes | Disabled by default (KB4019276)[[149]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-2008R1TLS1.2-194) | Disabled by default (KB4019276)[[149]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-2008R1TLS1.2-194) | No |
| [**SChannel 7 / 2008 R2**](https://en.wikipedia.org/wiki/SChannel)[**[216]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Windows7schannel-269) | Disabled by default | Disabled by default in MSIE 11 | Yes | Enabled by default by MSIE 11 | Enabled by default by MSIE 11 | No |
| [**SChannel 8 / 2012**](https://en.wikipedia.org/wiki/SChannel)[**[216]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Windows7schannel-269) | Disabled by default | Enabled by default | Yes | Disabled by default | Disabled by default | No |
| [**SChannel 8.1 / 2012 R2, 10 v1507 & v1511**](https://en.wikipedia.org/wiki/SChannel)[**[216]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Windows7schannel-269) | Disabled by default | Disabled by default in MSIE 11 | Yes | Yes | Yes | No |
| [**SChannel 10 v1607 / 2016**](https://en.wikipedia.org/wiki/SChannel)[**[159]**](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Windows10_v1607schannel-206) | No | Disabled by default | Yes | Yes | Yes | No |
| **Secure Transport OS X 10.2–10.8 / iOS 1–4** | Yes | Yes | Yes | No | No |  |
| **Secure Transport OS X 10.9–10.10 / iOS 5–8** | No[[c]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cnote_c_grp_protocollibrary-table) | Yes | Yes | Yes[[c]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cnote_c_grp_protocollibrary-table) | Yes[[c]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cnote_c_grp_protocollibrary-table) |  |
| **Secure Transport OS X 10.11 / iOS 9** | No | No[[c]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cnote_c_grp_protocollibrary-table) | Yes | Yes | Yes |  |
| [**Seed7**](https://en.wikipedia.org/wiki/Seed7)[**TLS/SSL Library**](http://seed7.sourceforge.net/libraries/tls.htm) | No | Yes | Yes | Yes | Yes |  |
| [**wolfSSL**](https://en.wikipedia.org/wiki/WolfSSL) **(previously CyaSSL)** | No | Disabled by default[[217]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-wolfSSL-3.6.6-270) | Yes | Yes | Yes | yes (draft version)[[218]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-wolfSSL_TLS_1.3_support-271) |
| **Implementation** | **SSL 2.0 (insecure)** | **SSL 3.0 (insecure)** | **TLS 1.0** | **TLS 1.1** | **TLS 1.2** | **TLS 1.3** |

 SSL 2.0 client hello is supported even though SSL 2.0 is not supported or is disabled because of the backward compatibilities.

  Server-side implementation of the SSL/TLS protocol still supports processing of received v2-compatible client hello messages.[[219]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-NSS-3.24-272)

1. 

Secure Transport: SSL 2.0 was discontinued in OS X 10.8. SSL 3.0 was discontinued in OS X 10.11 and iOS 9. TLS 1.1 and 1.2 are available on iOS 5.0 and later, and OS X 10.9 and later.[[220]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-273)

[[221]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-274)

A paper presented at the 2012 [ACM](https://en.wikipedia.org/wiki/Association_for_Computing_Machinery) [conference on computer and communications security](https://en.wikipedia.org/wiki/Computer_security_conference)[[222]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-275) showed that few applications used some of these SSL libraries correctly, leading to vulnerabilities. According to the authors

"the root cause of most of these vulnerabilities is the terrible design of the APIs to the underlying SSL libraries. Instead of expressing high-level security properties of network tunnels such as confidentiality and authentication, these APIs expose low-level details of the SSL protocol to application developers. As a consequence, developers often use SSL APIs incorrectly, misinterpreting and misunderstanding their manifold parameters, options, side effects, and return values."

**Other uses**

The [Simple Mail Transfer Protocol](https://en.wikipedia.org/wiki/Simple_Mail_Transfer_Protocol) (SMTP) can also be protected by TLS. These applications use [public key certificates](https://en.wikipedia.org/wiki/Public_key_certificate) to verify the identity of endpoints.

TLS can also be used for tunnelling an entire network stack to create a [VPN](https://en.wikipedia.org/wiki/Virtual_private_network), which is the case with [OpenVPN](https://en.wikipedia.org/wiki/OpenVPN) and [OpenConnect](https://en.wikipedia.org/wiki/OpenConnect). Many vendors have by now married TLS's encryption and authentication capabilities with authorization. There has also been substantial development since the late 1990s in creating client technology outside of Web-browsers, in order to enable support for client/server applications. Compared to traditional [IPsec](https://en.wikipedia.org/wiki/IPsec) VPN technologies, TLS has some inherent advantages in firewall and [NAT](https://en.wikipedia.org/wiki/Network_address_translation) traversal that make it easier to administer for large remote-access populations.

TLS is also a standard method for protecting [Session Initiation Protocol](https://en.wikipedia.org/wiki/Session_Initiation_Protocol) (SIP) application signaling. TLS can be used for providing authentication and encryption of the SIP signalling associated with [VoIP](https://en.wikipedia.org/wiki/Voice_over_Internet_Protocol) and other SIP-based applications.[[*citation needed*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)]

**Security**

**SSL 2.0**

SSL 2.0 was flawed in a variety of ways:[[223]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-276)

* Identical cryptographic keys were used for [message authentication](https://en.wikipedia.org/wiki/Message_authentication) and encryption. (In SSL 3.0, MAC secrets may be larger than encryption keys, so messages can remain tamper-resistant even if encryption keys are broken.[[5]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RFC6101-5))
* SSL 2.0 had a weak MAC construction that used the MD5 hash function with a secret prefix, making it vulnerable to [length extension attacks](https://en.wikipedia.org/wiki/Length_extension_attack).
* SSL 2.0 did not have any protection for the handshake, meaning a man-in-the-middle [downgrade attack](https://en.wikipedia.org/wiki/Downgrade_attack) could go undetected.
* SSL 2.0 used the TCP connection close to indicate the end of data. This meant that truncation attacks were possible: the attacker simply forges a TCP FIN, leaving the recipient unaware of an illegitimate end of data message (SSL 3.0 fixed this problem by having an explicit closure alert).
* SSL 2.0 assumed a single service and a fixed domain certificate, which clashed with the standard feature of virtual hosting in Web servers. This means that most websites were practically impaired from using SSL.

SSL 2.0 was disabled by default, beginning with [Internet Explorer 7](https://en.wikipedia.org/wiki/Internet_Explorer_7),[[224]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-277) [Mozilla Firefox](https://en.wikipedia.org/wiki/Mozilla_Firefox) 2,[[225]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-278) [Opera](https://en.wikipedia.org/wiki/Opera_(web_browser)) 9.5,[[226]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-279) and [Safari](https://en.wikipedia.org/wiki/Safari_(web_browser)). Support for SSL 2.0 (and weak [40-bit](https://en.wikipedia.org/wiki/40-bit_encryption) and 56-bit ciphers) was removed completely from Opera as of version 10.[[227]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-280)[[228]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-281)

**SSL 3.0**

SSL 3.0 improved upon SSL 2.0 by adding SHA-1–based ciphers and support for certificate authentication.

From a security standpoint, SSL 3.0 should be considered less desirable than TLS 1.0. The SSL 3.0 cipher suites have a weaker key derivation process; half of the master key that is established is fully dependent on the MD5 hash function, which is not resistant to collisions and is, therefore, not considered secure. Under TLS 1.0, the master key that is established depends on both MD5 and SHA-1 so its derivation process is not currently considered weak. It is for this reason that SSL 3.0 implementations cannot be validated under FIPS 140-2.[[229]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-282)

In October 2014, the vulnerability in the design of SSL 3.0 was reported, which makes CBC mode of operation with SSL 3.0 vulnerable to the padding attack (see [#POODLE attack](https://en.wikipedia.org/wiki/Transport_Layer_Security#POODLE_attack)).

**TLS**

TLS has a variety of security measures:

* Protection against a downgrade of the protocol to a previous (less secure) version or a weaker cipher suite.
* Numbering subsequent Application records with a sequence number and using this sequence number in the [message authentication codes](https://en.wikipedia.org/wiki/Message_authentication_code) (MACs).
* Using a message digest enhanced with a key (so only a key-holder can check the MAC). The [HMAC](https://en.wikipedia.org/wiki/HMAC) construction used by most TLS cipher suites is specified in [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [2104](https://tools.ietf.org/html/rfc2104) (SSL 3.0 used a different hash-based MAC).
* The message that ends the handshake ("Finished") sends a hash of all the exchanged handshake messages seen by both parties.
* The [pseudorandom](https://en.wikipedia.org/wiki/Pseudorandomness) function splits the input data in half and processes each one with a different hashing algorithm ([MD5](https://en.wikipedia.org/wiki/MD5) and [SHA-1](https://en.wikipedia.org/wiki/SHA-1)), then [XORs](https://en.wikipedia.org/wiki/Exclusive_or) them together to create the MAC. This provides protection even if one of these algorithms is found to be vulnerable.

**Attacks against TLS/SSL**

Significant attacks against TLS/SSL are listed below.

In February 2015, IETF issued an informational RFC[[230]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-283) summarizing the various known attacks against TLS/SSL.

**Renegotiation attack**

A vulnerability of the renegotiation procedure was discovered in August 2009 that can lead to plaintext injection attacks against SSL 3.0 and all current versions of TLS.[[231]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-284) For example, it allows an attacker who can hijack an https connection to splice their own requests into the beginning of the conversation the client has with the web server. The attacker can't actually decrypt the client–server communication, so it is different from a typical man-in-the-middle attack. A short-term fix is for web servers to stop allowing renegotiation, which typically will not require other changes unless [client certificate](https://en.wikipedia.org/wiki/Client_certificate) authentication is used. To fix the vulnerability, a renegotiation indication extension was proposed for TLS. It will require the client and server to include and verify information about previous handshakes in any renegotiation handshakes.[[232]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-285) This extension has become a proposed standard and has been assigned the number [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5746](https://tools.ietf.org/html/rfc5746). The RFC has been implemented by several libraries.[[233]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-286)[[234]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-287)[[235]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-288)

**Downgrade attacks: FREAK attack and Logjam attack**

Main articles: [Downgrade attack](https://en.wikipedia.org/wiki/Downgrade_attack), [FREAK](https://en.wikipedia.org/wiki/FREAK), and [Logjam (computer security)](https://en.wikipedia.org/wiki/Logjam_(computer_security))

A protocol [downgrade attack](https://en.wikipedia.org/wiki/Downgrade_attack) (also called a version rollback attack) tricks a web server into negotiating connections with previous versions of TLS (such as SSLv2) that have long since been abandoned as insecure.

Previous modifications to the original protocols, like **False Start**[[236]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-289) (adopted and enabled by Google Chrome[[237]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-290)) or **Snap Start**, reportedly introduced limited TLS protocol downgrade attacks[[238]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-291) or allowed modifications to the cipher suite list sent by the client to the server. In doing so, an attacker might succeed in influencing the cipher suite selection in an attempt to downgrade the cipher suite negotiated to use either a weaker symmetric encryption algorithm or a weaker key exchange.[[239]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-292) A paper presented at an [ACM](https://en.wikipedia.org/wiki/Association_for_Computing_Machinery) [conference on computer and communications security](https://en.wikipedia.org/wiki/Computer_security_conference) in 2012 demonstrated that the False Start extension was at risk: in certain circumstances it could allow an attacker to recover the encryption keys offline and to access the encrypted data.[[240]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-293)

Encryption downgrade attacks can force servers and clients to negotiate a connection using cryptographically weak keys. In 2014, a [man-in-the-middle](https://en.wikipedia.org/wiki/Man-in-the-middle) attack called FREAK was discovered affecting the [OpenSSL](https://en.wikipedia.org/wiki/OpenSSL) stack, the default [Android](https://en.wikipedia.org/wiki/Android_(operating_system)) web browser, and some [Safari](https://en.wikipedia.org/wiki/Safari_(web_browser)) browsers.[[241]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-294) The attack involved tricking servers into negotiating a TLS connection using cryptographically weak 512 bit encryption keys.

Logjam is a [security exploit](https://en.wikipedia.org/wiki/Security_exploit) discovered in May 2015 that exploits the option of using legacy ["export-grade"](https://en.wikipedia.org/wiki/Arms_Export_Control_Act) 512-bit [Diffie–Hellman](https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange) groups dating back to the 1990s.[[242]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-295) It forces susceptible servers to downgrade to cryptographically weak 512-bit Diffie–Hellman groups. An attacker can then deduce the keys the client and server determine using the [Diffie–Hellman key exchange](https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange).

**Cross-protocol attacks: DROWN**

Main article: [DROWN attack](https://en.wikipedia.org/wiki/DROWN_attack)

The [DROWN attack](https://en.wikipedia.org/wiki/DROWN_attack) is an exploit that attacks servers supporting contemporary SSL/TLS protocol suites by exploiting their support for the obsolete, insecure, SSLv2 protocol to leverage an attack on connections using up-to-date protocols that would otherwise be secure.[[243]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-296)[[244]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ars201603-297) DROWN exploits a vulnerability in the protocols used and the configuration of the server, rather than any specific implementation error. Full details of DROWN were announced in March 2016, together with a patch for the exploit. At that time, more than 81,000 of the top 1 million most popular websites were among the TLS protected websites that were vulnerable to the DROWN attack.[[244]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ars201603-297)

**BEAST attack**

On September 23, 2011 researchers Thai Duong and Juliano Rizzo demonstrated a proof of concept called **BEAST** (**Browser Exploit Against SSL/TLS**)[[245]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-DuongRizzo-298) using a [Java applet](https://en.wikipedia.org/wiki/Java_applet) to violate [same origin policy](https://en.wikipedia.org/wiki/Same_origin_policy) constraints, for a long-known [cipher block chaining](https://en.wikipedia.org/wiki/Cipher_block_chaining) (CBC) vulnerability in TLS 1.0:[[246]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-DanGoodin-299)[[247]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-combinator-300) an attacker observing 2 consecutive ciphertext blocks C0, C1 can test if the plaintext block P1 is equal to x by choosing the next plaintext block P2 = x ⊕ {\displaystyle \oplus } C0 ⊕ {\displaystyle \oplus } C1; as per CBC operation, C2 = E(C1 ⊕ {\displaystyle \oplus } P2) = E(C1 ⊕ {\displaystyle \oplus } x ⊕ {\displaystyle \oplus } C0 ⊕ {\displaystyle \oplus } C1) = E(C0 ⊕ {\displaystyle \oplus } x), which will be equal to C1 if x = P1. Practical [exploits](https://en.wikipedia.org/wiki/Exploit_(computer_security)) had not been previously demonstrated for this [vulnerability](https://en.wikipedia.org/wiki/Vulnerability_(computing)), which was originally discovered by [Phillip Rogaway](https://en.wikipedia.org/wiki/Phillip_Rogaway)[[248]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-301) in 2002. The vulnerability of the attack had been fixed with TLS 1.1 in 2006, but TLS 1.1 had not seen wide adoption prior to this attack demonstration.

[RC4](https://en.wikipedia.org/wiki/RC4) as a stream cipher is immune to BEAST attack. Therefore, RC4 was widely used as a way to mitigate BEAST attack on the server side. However, in 2013, researchers found more weaknesses in RC4. Thereafter enabling RC4 on server side was no longer recommended.[[249]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-302)

Chrome and Firefox themselves are not vulnerable to BEAST attack,[[79]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ChromeBEAST-102)[[99]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-FirefoxBEAST-130) however, Mozilla updated their [NSS](https://en.wikipedia.org/wiki/Network_Security_Services) libraries to mitigate BEAST-like [attacks](https://en.wikipedia.org/wiki/Attack_(computing)). NSS is used by [Mozilla Firefox](https://en.wikipedia.org/wiki/Mozilla_Firefox) and [Google Chrome](https://en.wikipedia.org/wiki/Google_Chrome) to implement SSL. Some [web servers](https://en.wikipedia.org/wiki/Web_server) that have a broken implementation of the SSL specification may stop working as a result.[[250]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-303)

[Microsoft](https://en.wikipedia.org/wiki/Microsoft) released Security Bulletin MS12-006 on January 10, 2012, which fixed the BEAST vulnerability by changing the way that the Windows Secure Channel ([SChannel](https://en.wikipedia.org/wiki/SChannel)) component transmits encrypted network packets from the server end.[[251]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-304) Users of Internet Explorer (prior to version 11) that run on older versions of Windows ([Windows 7](https://en.wikipedia.org/wiki/Windows_7), [Windows 8](https://en.wikipedia.org/wiki/Windows_8) and [Windows Server 2008 R2](https://en.wikipedia.org/wiki/Windows_Server_2008)) can restrict use of TLS to 1.1 or higher.

[Apple](https://en.wikipedia.org/wiki/Apple_Inc.) fixed BEAST vulnerability by implementing 1/n-1 split and turning it on by default in [OS X Mavericks](https://en.wikipedia.org/wiki/OS_X_Mavericks), released on October 22, 2013.[[252]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-305)

**CRIME and BREACH attacks**

Main articles: [CRIME](https://en.wikipedia.org/wiki/CRIME) and [BREACH](https://en.wikipedia.org/wiki/BREACH)

The authors of the BEAST attack are also the creators of the later [CRIME](https://en.wikipedia.org/wiki/CRIME) attack, which can allow an attacker to recover the content of web cookies when [data compression](https://en.wikipedia.org/wiki/Data_compression) is used along with TLS.[[253]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-306)[[254]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-307) When used to recover the content of secret [authentication cookies](https://en.wikipedia.org/wiki/Authentication_cookie), it allows an attacker to perform [session hijacking](https://en.wikipedia.org/wiki/Session_hijacking) on an authenticated web session.

While the CRIME attack was presented as a general attack that could work effectively against a large number of protocols, including but not limited to TLS, and application-layer protocols such as [SPDY](https://en.wikipedia.org/wiki/SPDY) or [HTTP](https://en.wikipedia.org/wiki/HTTP), only exploits against TLS and SPDY were demonstrated and largely mitigated in browsers and servers. The CRIME exploit against [HTTP compression](https://en.wikipedia.org/wiki/HTTP_compression) has not been mitigated at all, even though the authors of CRIME have warned that this vulnerability might be even more widespread than SPDY and TLS compression combined. In 2013 a new instance of the CRIME attack against HTTP compression, dubbed [BREACH](https://en.wikipedia.org/wiki/BREACH), was announced. Based on the CRIME attack a BREACH attack can extract login tokens, email addresses or other sensitive information from TLS encrypted web traffic in as little as 30 seconds (depending on the number of bytes to be extracted), provided the attacker tricks the victim into visiting a malicious web link or is able to inject content into valid pages the user is visiting (ex: a wireless network under the control of the attacker).[[255]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Gooin20130801-308) All versions of TLS and SSL are at risk from BREACH regardless of the encryption algorithm or cipher used.[[256]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-309) Unlike previous instances of CRIME, which can be successfully defended against by turning off TLS compression or SPDY header compression, BREACH exploits HTTP compression which cannot realistically be turned off, as virtually all web servers rely upon it to improve data transmission speeds for users.[[255]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Gooin20130801-308) This is a known limitation of TLS as it is susceptible to [chosen-plaintext attack](https://en.wikipedia.org/wiki/Chosen-plaintext_attack) against the application-layer data it was meant to protect.

**Timing attacks on padding**

Earlier TLS versions were vulnerable against the [padding oracle attack](https://en.wikipedia.org/wiki/Padding_oracle_attack) discovered in 2002. A novel variant, called the [Lucky Thirteen attack](https://en.wikipedia.org/wiki/Lucky_Thirteen_attack), was published in 2013.

Some experts[[62]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-best-practices-69) also recommended avoiding [Triple-DES](https://en.wikipedia.org/wiki/Triple_DES) CBC. Since the last supported ciphers developed to support any program using [Windows XP](https://en.wikipedia.org/wiki/Windows_XP)'s SSL/TLS library like Internet Explorer on Windows XP are [RC4](https://en.wikipedia.org/wiki/RC4) and Triple-DES, and since RC4 is now deprecated (see discussion of [RC4 attacks](https://en.wikipedia.org/wiki/Transport_Layer_Security#RC4)), this makes it difficult to support any version of SSL for any program using this library on XP.

A fix was released as the Encrypt-then-MAC extension to the TLS specification, released as [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7366](https://tools.ietf.org/html/rfc7366).[[257]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-310) The Lucky Thirteen attack can be mitigated in TLS 1.2 by using only AES\_GCM ciphers; AES\_CBC remains vulnerable.[[*citation needed*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)]

**POODLE attack**

Main article: [POODLE](https://en.wikipedia.org/wiki/POODLE)

On October 14, 2014, Google researchers published a vulnerability in the design of SSL 3.0, which makes [CBC mode of operation](https://en.wikipedia.org/wiki/CBC_mode_of_operation) with SSL 3.0 vulnerable to a [padding attack](https://en.wikipedia.org/wiki/Padding_oracle_attack) ([CVE](https://en.wikipedia.org/wiki/CVE_(identifier))-[2014-3566](https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2014-3566)). They named this attack **POODLE** (**Padding Oracle On Downgraded Legacy Encryption**). On average, attackers only need to make 256 SSL 3.0 requests to reveal one byte of encrypted messages.[[69]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-poodle_pdf-81)

Although this vulnerability only exists in SSL 3.0 and most clients and servers support TLS 1.0 and above, all major browsers voluntarily downgrade to SSL 3.0 if the handshakes with newer versions of TLS fail unless they provide the option for a user or administrator to disable SSL 3.0 and the user or administrator does so[[*citation needed*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)]. Therefore, the man-in-the-middle can first conduct a [version rollback attack](https://en.wikipedia.org/wiki/Version_rollback_attack) and then exploit this vulnerability.[[69]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-poodle_pdf-81)

In general, graceful security degradation for the sake of interoperability is difficult to carry out in a way that cannot be exploited. This is challenging especially in domains where fragmentation is high.[[258]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-311)

On December 8, 2014, a variant of POODLE was announced that impacts TLS implementations that do not properly enforce padding byte requirements.[[259]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-poodleagain-312)

**RC4 attacks**

Main article: [RC4 § Security](https://en.wikipedia.org/wiki/RC4#Security)

Despite the existence of attacks on [RC4](https://en.wikipedia.org/wiki/RC4) that broke its security, cipher suites in SSL and TLS that were based on RC4 were still considered secure prior to 2013 based on the way in which they were used in SSL and TLS. In 2011, the RC4 suite was actually recommended as a work around for the [BEAST](https://en.wikipedia.org/wiki/BEAST_(computer_security)) attack.[[260]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-313) New forms of attack disclosed in March 2013 conclusively demonstrated the feasibility of breaking RC4 in TLS, suggesting it was not a good workaround for BEAST.[[68]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-community.qualys-80) An attack scenario was proposed by AlFardan, Bernstein, Paterson, Poettering and Schuldt that used newly discovered statistical biases in the RC4 key table[[261]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-314) to recover parts of the plaintext with a large number of TLS encryptions.[[262]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-315)[[263]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-316) An attack on RC4 in TLS and SSL that requires 13 × 220 encryptions to break RC4 was unveiled on 8 July 2013 and later described as "feasible" in the accompanying presentation at a [USENIX](https://en.wikipedia.org/wiki/USENIX) Security Symposium in August 2013.[[264]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-317)[[265]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-318) In July 2015, subsequent improvements in the attack make it increasingly practical to defeat the security of RC4-encrypted TLS.[[266]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-319)

As many modern browsers have been designed to defeat BEAST attacks (except Safari for Mac OS X 10.7 or earlier, for iOS 6 or earlier, and for Windows; see [#Web browsers](https://en.wikipedia.org/wiki/Transport_Layer_Security#Web_browsers)), RC4 is no longer a good choice for TLS 1.0. The CBC ciphers which were affected by the BEAST attack in the past have become a more popular choice for protection.[[62]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-best-practices-69) Mozilla and Microsoft recommend disabling RC4 where possible.[[267]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-320)[[268]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-321) [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7465](https://tools.ietf.org/html/rfc7465) prohibits the use of RC4 cipher suites in all versions of TLS.

On September 1, 2015, Microsoft, Google and Mozilla announced that RC4 cipher suites would be disabled by default in their browsers ([Microsoft Edge](https://en.wikipedia.org/wiki/Microsoft_Edge), [Internet Explorer 11](https://en.wikipedia.org/wiki/Internet_Explorer_11) on Windows 7/8.1/10, [Firefox](https://en.wikipedia.org/wiki/Firefox), and [Chrome](https://en.wikipedia.org/wiki/Google_Chrome)) in early 2016.[[269]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-322)[[270]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-323)[[271]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-324)

**Truncation attack**

A TLS (logout) truncation attack blocks a victim's account logout requests so that the user unknowingly remains logged into a web service. When the request to sign out is sent, the attacker injects an unencrypted [TCP](https://en.wikipedia.org/wiki/Transmission_Control_Protocol) FIN message (no more data from sender) to close the connection. The server therefore doesn't receive the logout request and is unaware of the abnormal termination.[[272]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-register20130801-325)

Published in July 2013,[[273]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-326)[[274]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-327) the attack causes web services such as [Gmail](https://en.wikipedia.org/wiki/Gmail) and [Hotmail](https://en.wikipedia.org/wiki/Outlook.com) to display a page that informs the user that they have successfully signed-out, while ensuring that the user's browser maintains authorization with the service, allowing an attacker with subsequent access to the browser to access and take over control of the user's logged-in account. The attack does not rely on installing malware on the victim's computer; attackers need only place themselves between the victim and the web server (e.g., by setting up a rogue wireless hotspot).[[272]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-register20130801-325) This vulnerability also requires access to the victim's computer. Another possibility is when using FTP the data connection can have a false FIN in the data stream, and if the protocol rules for exchanging close\_notify alerts is not adhered to a file can be truncated.

**Unholy PAC attack**

This attack, discovered in mid-2016, exploits weaknesses in the [Web Proxy Autodiscovery Protocol](https://en.wikipedia.org/wiki/Web_Proxy_Autodiscovery_Protocol) (WPAD) to expose the URL that a web user is attempting to reach via a TLS-enabled web link.[[275]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-328) Disclosure of a URL can violate a user's privacy, not only because of the website accessed, but also because URLs are sometimes used to authenticate users. Document sharing services, such as those offered by Google and Dropbox, also work by sending a user a security token that's included in the URL. An attacker who obtains such URLs may be able to gain full access to a victim's account or data.

The exploit works against almost all browsers and operating systems.

**Sweet32 attack**

The Sweet32 attack breaks all 64-bit block ciphers used in CBC mode as used in TLS by exploiting a [birthday attack](https://en.wikipedia.org/wiki/Birthday_attack) and either a [man-in-the-middle attack](https://en.wikipedia.org/wiki/Man-in-the-middle_attack) or injection of a malicious [JavaScript](https://en.wikipedia.org/wiki/JavaScript) into a web page. The purpose of the man-in-the-middle attack or the JavaScript injection is to allow the attacker to capture enough traffic to mount a birthday attack.[[276]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-329)

**Implementation errors: Heartbleed bug, BERserk attack, Cloudflare bug**

Main articles: [Heartbleed](https://en.wikipedia.org/wiki/Heartbleed) and [Cloudbleed](https://en.wikipedia.org/wiki/Cloudbleed)

The [Heartbleed](https://en.wikipedia.org/wiki/Heartbleed) bug is a serious vulnerability specific to the implementation of SSL/TLS in the popular [OpenSSL](https://en.wikipedia.org/wiki/OpenSSL) cryptographic software library, affecting versions 1.0.1 to 1.0.1f. This weakness, reported in April 2014, allows attackers to steal [private keys](https://en.wikipedia.org/wiki/Public-key_cryptography) from servers that should normally be protected.[[277]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-330) The Heartbleed bug allows anyone on the Internet to read the memory of the systems protected by the vulnerable versions of the OpenSSL software. This compromises the secret private keys associated with the [public certificates](https://en.wikipedia.org/wiki/X.509) used to identify the service providers and to encrypt the traffic, the names and passwords of the users and the actual content. This allows attackers to eavesdrop on communications, steal data directly from the services and users and to impersonate services and users.[[278]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-331) The vulnerability is caused by a [buffer over-read](https://en.wikipedia.org/wiki/Buffer_over-read) bug in the OpenSSL software, rather than a defect in the SSL or TLS protocol specification.

In September 2014, a variant of Daniel Bleichenbacher's PKCS#1 v1.5 RSA Signature Forgery vulnerability[[279]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-332) was announced by Intel Security Advanced Threat Research. This attack, dubbed BERserk, is a result of incomplete ASN.1 length decoding of public key signatures in some SSL implementations, and allows a man-in-the-middle attack by forging a public key signature.[[280]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-333)

In February 2015, after media reported the hidden pre-installation of [Superfish](https://en.wikipedia.org/wiki/Superfish) adware on some Lenovo notebooks,[[281]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-334) a researcher found a trusted root certificate on affected Lenovo machines to be insecure, as the keys could easily be accessed using the company name, Komodia, as a passphrase.[[282]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-335) The Komodia library was designed to intercept client-side TLS/SSL traffic for parental control and surveillance, but it was also used in numerous adware programs, including Superfish, that were often surreptitiously installed unbeknownst to the computer user. In turn, these [potentially unwanted programs](https://en.wikipedia.org/wiki/Potentially_unwanted_program) installed the corrupt root certificate, allowing attackers to completely control web traffic and confirm false websites as authentic.

In May 2016, it was reported that dozens of Danish HTTPS-protected websites belonging to [Visa Inc.](https://en.wikipedia.org/wiki/Visa_Inc.) were vulnerable to attacks allowing hackers to inject malicious code and forged content into the browsers of visitors.[[283]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-forbidden-336) The attacks worked because the TLS implementation used on the affected servers incorrectly reused random numbers ([nonces](https://en.wikipedia.org/wiki/Cryptographic_nonce)) that are intended be used only once, ensuring that each [TLS handshake](https://en.wikipedia.org/wiki/Transport_Layer_Security#TLS_handshake) is unique.[[283]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-forbidden-336)

In February 2017, an implementation error caused by a single mistyped character in code used to parse HTML created a buffer overflow error on [Cloudflare](https://en.wikipedia.org/wiki/Cloudflare) servers. Similar in its effects to the Heartbleed bug discovered in 2014, this overflow error, widely known as [Cloudbleed](https://en.wikipedia.org/wiki/Cloudbleed), allowed unauthorized third parties to read data in the memory of programs running on the servers—data that should otherwise have been protected by TLS.[[284]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-337)

**Survey of websites vulnerable to attacks**

As of August 2019, Trustworthy Internet Movement estimate the ratio of websites that are vulnerable to TLS attacks.[[67]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-trustworthy_ssl_pulse-79)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Survey of the TLS vulnerabilities of the most popular websites | | | | |
| **Attacks** | **Security** | | | |
| **Insecure** | **Depends** | **Secure** | **Other** |
| [**Renegotiation attack**](https://en.wikipedia.org/wiki/Transport_Layer_Security#Renegotiation_attack) | 0.3%  support insecure renegotiation | 0.1%  support both | 98.4%  support secure renegotiation | 1.1%  no support |
| [**RC4 attacks**](https://en.wikipedia.org/wiki/Transport_Layer_Security#RC4_attacks) | 1.2% support RC4 suites used with modern browsers | 12.1% support some RC4 suites | 86.7% no support | N/A |
| [**TLS Compression (CRIME attack)**](https://en.wikipedia.org/wiki/Transport_Layer_Security#CRIME_attack) | 0.6% vulnerable | N/A | N/A | N/A |
| [**Heartbleed**](https://en.wikipedia.org/wiki/Transport_Layer_Security#Heartbleed) | <0.1% vulnerable | N/A | N/A | N/A |
| [**ChangeCipherSpec injection attack**](https://en.wikipedia.org/wiki/CVE-2014-0224) | 0.2% vulnerable and exploitable | 1.2%  vulnerable, not exploitable | 96.9%  not vulnerable | 1.7%  unknown |
| [**POODLE attack against TLS**](https://en.wikipedia.org/wiki/Transport_Layer_Security#POODLE_attack) **(Original POODLE against SSL 3.0 is not included)** | 0.3% vulnerable and exploitable | N/A | 99.5% not vulnerable | 0.2%  unknown |
| [**Protocol downgrade**](https://en.wikipedia.org/wiki/Transport_Layer_Security#Downgrade_attacks) | 11.3% Downgrade defence not supported | N/A | 71.6% Downgrade defence supported | 17.0%  unknown |

**Forward secrecy**

Main article: [Forward secrecy](https://en.wikipedia.org/wiki/Forward_secrecy)

[Forward secrecy](https://en.wikipedia.org/wiki/Forward_secrecy) is a property of cryptographic systems which ensures that a session key derived from a set of public and private keys will not be compromised if one of the private keys is compromised in the future.[[285]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-338) Without forward secrecy, if the server's private key is compromised, not only will all future TLS-encrypted sessions using that server certificate be compromised, but also any past sessions that used it as well (provided of course that these past sessions were intercepted and stored at the time of transmission).[[286]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-339) An implementation of TLS can provide forward secrecy by requiring the use of ephemeral [Diffie–Hellman key exchange](https://en.wikipedia.org/wiki/Diffie%E2%80%93Hellman_key_exchange) to establish session keys, and some notable TLS implementations do so exclusively: e.g., [Gmail](https://en.wikipedia.org/wiki/Gmail) and other Google HTTPS services that use [OpenSSL](https://en.wikipedia.org/wiki/OpenSSL).[[287]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-340) However, many clients and servers supporting TLS (including browsers and web servers) are not configured to implement such restrictions.[[288]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-341)[[289]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-342) In practice, unless a web service uses Diffie–Hellman key exchange to implement forward secrecy, all of the encrypted web traffic to and from that service can be decrypted by a third party if it obtains the server's master (private) key; e.g., by means of a court order.[[290]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-343)

Even where Diffie–Hellman key exchange is implemented, server-side session management mechanisms can impact forward secrecy. The use of [TLS session tickets](https://en.wikipedia.org/wiki/Transport_Layer_Security#Resumed_TLS_handshake) (a TLS extension) causes the session to be protected by AES128-CBC-SHA256 regardless of any other negotiated TLS parameters, including forward secrecy ciphersuites, and the long-lived TLS session ticket keys defeat the attempt to implement forward secrecy.[[291]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-botchingpfs-344)[[292]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ticketsecwp-345)[[293]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ticketsecslides-346) Stanford University research in 2014 also found that of 473,802 TLS servers surveyed, 82.9% of the servers deploying ephemeral Diffie–Hellman (DHE) key exchange to support forward secrecy were using weak Diffie–Hellman parameters. These weak parameter choices could potentially compromise the effectiveness of the forward secrecy that the servers sought to provide.[[294]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-347)

Since late 2011, Google has provided forward secrecy with TLS by default to users of its [Gmail](https://en.wikipedia.org/wiki/Gmail) service, along with [Google Docs](https://en.wikipedia.org/wiki/Google_Docs) and encrypted search, among other services.[[295]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-googleLongTerm-348) Since November 2013, [Twitter](https://en.wikipedia.org/wiki/Twitter) has provided forward secrecy with TLS to users of its service.[[296]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-349) As of August 2019, about 80% of TLS-enabled websites are configured to use cipher suites that provide forward secrecy to most web browsers.[[67]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-trustworthy_ssl_pulse-79)

**TLS interception**

TLS interception (or [HTTPS](https://en.wikipedia.org/wiki/HTTPS) interception if applied particularly to that protocol) is the practice of intercepting an encrypted data stream in order to decrypt it, read and possibly manipulate it, and then re-encrypt it and send the data on its way again. This is done by way of a "[transparent proxy](https://en.wikipedia.org/wiki/Transparent_proxy)": the interception software terminates the incoming TLS connection, inspects the HTTP plaintext, and then creates a new TLS connection to the destination.[[297]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Durumeric_et_al_2017-350)

TLS / HTTPS interception is used as an [information security](https://en.wikipedia.org/wiki/Information_security) measure by network operators in order to be able to scan for and protect against the intrusion of malicious content into the network, such as [computer viruses](https://en.wikipedia.org/wiki/Computer_virus) and other [malware](https://en.wikipedia.org/wiki/Malware).[[297]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Durumeric_et_al_2017-350) Such content could otherwise not be detected as long as it is protected by encryption, which is increasingly the case as a result of the routine use of HTTPS and other secure protocols.

A significant drawback of TLS / HTTPS interception is that it introduces new security risks of its own. Because it provides a point where network traffic is available unencrypted, attackers have an incentive to attack this point in particular in order to gain access to otherwise secure content. The interception also allows the network operator, or persons who gain access to its interception system, to perform [man-in-the-middle attacks](https://en.wikipedia.org/wiki/Man-in-the-middle_attack) against network users. A 2017 study found that "HTTPS interception has become startlingly widespread, and that interception products as a class have a dramatically negative impact on connection security".[[297]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-Durumeric_et_al_2017-350)

**Protocol details**

The TLS protocol exchanges *records*, which encapsulate the data to be exchanged in a specific format (see below). Each record can be compressed, padded, appended with a [message authentication code](https://en.wikipedia.org/wiki/Message_authentication_code) (MAC), or encrypted, all depending on the state of the connection. Each record has a *content type* field that designates the type of data encapsulated, a length field and a TLS version field. The data encapsulated may be control or procedural messages of the TLS itself, or simply the application data needed to be transferred by TLS. The specifications (cipher suite, keys etc.) required to exchange application data by TLS, are agreed upon in the "TLS handshake" between the client requesting the data and the server responding to requests. The protocol therefore defines both the structure of payloads transferred in TLS and the procedure to establish and monitor the transfer.

**TLS handshake**

When the connection starts, the record encapsulates a "control" protocol – the handshake messaging protocol (*content type* 22). This protocol is used to exchange all the information required by both sides for the exchange of the actual application data by TLS. It defines the format of messages and the order of their exchange. These may vary according to the demands of the client and server – i.e., there are several possible procedures to set up the connection. This initial exchange results in a successful TLS connection (both parties ready to transfer application data with TLS) or an alert message (as specified below).

**Basic TLS handshake**

A typical connection example follows, illustrating a [handshake](https://en.wikipedia.org/wiki/Handshaking) where the server (but not the client) is authenticated by its certificate:

1. Negotiation phase:
   * A client sends a **ClientHello** message specifying the highest TLS protocol version it supports, a random number, a list of suggested [cipher suites](https://en.wikipedia.org/wiki/Cipher_suite) and suggested compression methods. If the client is attempting to perform a resumed handshake, it may send a *session ID*. If the client can use [Application-Layer Protocol Negotiation](https://en.wikipedia.org/wiki/Application-Layer_Protocol_Negotiation), it may include a list of supported application [protocols](https://en.wikipedia.org/wiki/Communications_protocol), such as [HTTP/2](https://en.wikipedia.org/wiki/HTTP/2).
   * The server responds with a **ServerHello** message, containing the chosen protocol version, a random number, cipher suite and compression method from the choices offered by the client. To confirm or allow resumed handshakes the server may send a *session ID*. The chosen protocol version should be the highest that both the client and server support. For example, if the client supports TLS version 1.1 and the server supports version 1.2, version 1.1 should be selected; version 1.2 should not be selected.
   * The server sends its **Certificate** message (depending on the selected cipher suite, this may be omitted by the server).[[298]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-openpgp-351)
   * The server sends its **ServerKeyExchange** message (depending on the selected cipher suite, this may be omitted by the server). This message is sent for all DHE and DH\_anon cipher suites.[[2]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RFC5246-2)
   * The server sends a **ServerHelloDone** message, indicating it is done with handshake negotiation.
   * The client responds with a **ClientKeyExchange** message, which may contain a *PreMasterSecret*, public key, or nothing. (Again, this depends on the selected cipher.) This *PreMasterSecret* is encrypted using the public key of the server certificate.
   * The client and server then use the random numbers and *PreMasterSecret* to compute a common secret, called the "master secret". All other key data for this connection is derived from this master secret (and the client- and server-generated random values), which is passed through a carefully designed [pseudorandom](https://en.wikipedia.org/wiki/Pseudorandomness) function.
2. The client now sends a **ChangeCipherSpec** record, essentially telling the server, "Everything I tell you from now on will be authenticated (and encrypted if encryption parameters were present in the server certificate)." The ChangeCipherSpec is itself a record-level protocol with content type of 20.
   * The client sends an authenticated and encrypted **Finished** message, containing a hash and MAC over the previous handshake messages.
   * The server will attempt to decrypt the client's *Finished* message and verify the hash and MAC. If the decryption or verification fails, the handshake is considered to have failed and the connection should be torn down.
3. Finally, the server sends a **ChangeCipherSpec**, telling the client, "Everything I tell you from now on will be authenticated (and encrypted, if encryption was negotiated)."
   * The server sends its authenticated and encrypted **Finished** message.
   * The client performs the same decryption and verification procedure as the server did in the previous step.
4. Application phase: at this point, the "handshake" is complete and the application protocol is enabled, with content type of 23. Application messages exchanged between client and server will also be authenticated and optionally encrypted exactly like in their *Finished* message. Otherwise, the content type will return 25 and the client will not authenticate.

**Client-authenticated TLS handshake**

The following *full* example shows a client being authenticated (in addition to the server as in the example above) via TLS using certificates exchanged between both peers.

1. Negotiation Phase:
   * A client sends a **ClientHello** message specifying the highest TLS protocol version it supports, a random number, a list of suggested cipher suites and compression methods.
   * The server responds with a **ServerHello** message, containing the chosen protocol version, a random number, cipher suite and compression method from the choices offered by the client. The server may also send a *session id* as part of the message to perform a resumed handshake.
   * The server sends its **Certificate** message (depending on the selected cipher suite, this may be omitted by the server).[[298]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-openpgp-351)
   * The server sends its **ServerKeyExchange** message (depending on the selected cipher suite, this may be omitted by the server). This message is sent for all DHE and DH\_anon ciphersuites.[[2]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-RFC5246-2)
   * The server sends a **CertificateRequest** message, to request a certificate from the client so that the connection can be [mutually authenticated](https://en.wikipedia.org/wiki/Mutually_authenticated).
   * The server sends a **ServerHelloDone** message, indicating it is done with handshake negotiation.
   * The client responds with a **Certificate** message, which contains the client's certificate.
   * The client sends a **ClientKeyExchange** message, which may contain a *PreMasterSecret*, public key, or nothing. (Again, this depends on the selected cipher.) This *PreMasterSecret* is encrypted using the public key of the server certificate.
   * The client sends a **CertificateVerify** message, which is a signature over the previous handshake messages using the client's certificate's private key. This signature can be verified by using the client's certificate's public key. This lets the server know that the client has access to the private key of the certificate and thus owns the certificate.
   * The client and server then use the random numbers and *PreMasterSecret* to compute a common secret, called the "master secret". All other key data for this connection is derived from this master secret (and the client- and server-generated random values), which is passed through a carefully designed pseudorandom function.
2. The client now sends a **ChangeCipherSpec** record, essentially telling the server, "Everything I tell you from now on will be authenticated (and encrypted if encryption was negotiated). " The ChangeCipherSpec is itself a record-level protocol and has type 20 and not 22.
   * Finally, the client sends an encrypted **Finished** message, containing a hash and MAC over the previous handshake messages.
   * The server will attempt to decrypt the client's *Finished* message and verify the hash and MAC. If the decryption or verification fails, the handshake is considered to have failed and the connection should be torn down.
3. Finally, the server sends a **ChangeCipherSpec**, telling the client, "Everything I tell you from now on will be authenticated (and encrypted if encryption was negotiated). "
   * The server sends its own encrypted **Finished** message.
   * The client performs the same decryption and verification procedure as the server did in the previous step.
4. Application phase: at this point, the "handshake" is complete and the application protocol is enabled, with content type of 23. Application messages exchanged between client and server will also be encrypted exactly like in their *Finished* message.

**Resumed TLS handshake**

Public key operations (e.g., RSA) are relatively expensive in terms of computational power. TLS provides a secure shortcut in the handshake mechanism to avoid these operations: resumed sessions. Resumed sessions are implemented using session IDs or session tickets.

Apart from the performance benefit, resumed sessions can also be used for [single sign-on](https://en.wikipedia.org/wiki/Single_sign-on), as it guarantees that both the original session and any resumed session originate from the same client. This is of particular importance for the [FTP over TLS/SSL](https://en.wikipedia.org/wiki/FTPS) protocol, which would otherwise suffer from a man-in-the-middle attack in which an attacker could intercept the contents of the secondary data connections.[[299]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-352)

**TLS 1.3 handshake**

The TLS 1.3 handshake was condensed to only one round trip compared to the two round trips required in previous versions of TLS/SSL.

First the client sends a clientHello message to the server that contains a list of supported ciphers in order of the client's preference and makes a guess on what key algorithm will be used so that it can send a secret key to share if needed. By making a guess at what key algorithm will be used, the server eliminates a round trip. After receiving the clientHello, the server sends a serverHello with its key, a certificate, the chosen cipher suite and the finished message.

After the client receives the server's finished message, it now is coordinated with the server on which cipher suite to use.[[1]](https://blog.cloudflare.com/tls-1-3-overview-and-q-and-a/)

**Session IDs**

In an ordinary *full* handshake, the server sends a *session id* as part of the **ServerHello** message. The client associates this *session id* with the server's IP address and TCP port, so that when the client connects again to that server, it can use the *session id* to shortcut the handshake. In the server, the *session id* maps to the cryptographic parameters previously negotiated, specifically the "master secret". Both sides must have the same "master secret" or the resumed handshake will fail (this prevents an eavesdropper from using a *session id*). The random data in the **ClientHello** and **ServerHello** messages virtually guarantee that the generated connection keys will be different from in the previous connection. In the RFCs, this type of handshake is called an *abbreviated* handshake. It is also described in the literature as a *restart* handshake.

1. Negotiation phase:
   * A client sends a **ClientHello** message specifying the highest TLS protocol version it supports, a random number, a list of suggested cipher suites and compression methods. Included in the message is the *session id* from the previous TLS connection.
   * The server responds with a **ServerHello** message, containing the chosen protocol version, a random number, cipher suite and compression method from the choices offered by the client. If the server recognizes the *session id* sent by the client, it responds with the same *session id*. The client uses this to recognize that a resumed handshake is being performed. If the server does not recognize the *session id* sent by the client, it sends a different value for its *session id*. This tells the client that a resumed handshake will not be performed. At this point, both the client and server have the "master secret" and random data to generate the key data to be used for this connection.
2. The server now sends a **ChangeCipherSpec** record, essentially telling the client, "Everything I tell you from now on will be encrypted." The ChangeCipherSpec is itself a record-level protocol and has type 20 and not 22.
   * Finally, the server sends an encrypted **Finished** message, containing a hash and MAC over the previous handshake messages.
   * The client will attempt to decrypt the server's *Finished* message and verify the hash and MAC. If the decryption or verification fails, the handshake is considered to have failed and the connection should be torn down.
3. Finally, the client sends a **ChangeCipherSpec**, telling the server, "Everything I tell you from now on will be encrypted. "
   * The client sends its own encrypted **Finished** message.
   * The server performs the same decryption and verification procedure as the client did in the previous step.
4. Application phase: at this point, the "handshake" is complete and the application protocol is enabled, with content type of 23. Application messages exchanged between client and server will also be encrypted exactly like in their *Finished* message.

**Session tickets**

[RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5077](https://tools.ietf.org/html/rfc5077) extends TLS via use of session tickets, instead of session IDs. It defines a way to resume a TLS session without requiring that session-specific state is stored at the TLS server.

When using session tickets, the TLS server stores its session-specific state in a session ticket and sends the session ticket to the TLS client for storing. The client resumes a TLS session by sending the session ticket to the server, and the server resumes the TLS session according to the session-specific state in the ticket. The session ticket is encrypted and authenticated by the server, and the server verifies its validity before using its contents.

One particular weakness of this method with [OpenSSL](https://en.wikipedia.org/wiki/OpenSSL) is that it always limits encryption and authentication security of the transmitted TLS session ticket to AES128-CBC-SHA256, no matter what other TLS parameters were negotiated for the actual TLS session.[[292]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ticketsecwp-345) This means that the state information (the TLS session ticket) is not as well protected as the TLS session itself. Of particular concern is OpenSSL's storage of the keys in an application-wide context (SSL\_CTX), i.e. for the life of the application, and not allowing for re-keying of the AES128-CBC-SHA256 TLS session tickets without resetting the application-wide OpenSSL context (which is uncommon, error-prone and often requires manual administrative intervention).[[293]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-ticketsecslides-346)[[291]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-botchingpfs-344)

**TLS record**

This is the general format of all TLS records.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **+** | **Byte +0** | **Byte +1** | **Byte +2** | **Byte +3** |
| **Byte 0** | Content type |  | | |
| **Bytes 1..4** | Legacy version | | Length | |
| *(Major)* | *(Minor)* | *(bits 15..8)* | *(bits 7..0)* |
| **Bytes 5..(*m*−1)** | Protocol message(s) | | | |
| **Bytes *m*..(*p*−1)** | [MAC](https://en.wikipedia.org/wiki/Message_authentication_code) (optional) | | | |
| **Bytes *p*..(*q*−1)** | Padding (block ciphers only) | | | |

Content type

This field identifies the Record Layer Protocol Type contained in this Record.

|  |  |  |
| --- | --- | --- |
| Content types | | |
| **Hex** | **Dec** | **Type** |
| 0x14 | 20 | ChangeCipherSpec |
| 0x15 | 21 | Alert |
| 0x16 | 22 | Handshake |
| 0x17 | 23 | Application |
| 0x18 | 24 | Heartbeat |

Legacy version

This field identifies the major and minor version of TLS prior to TLS 1.3 for the contained message. For a ClientHello message, this need not be the *highest* version supported by the client. For TLS 1.3 and later, this must to be set 0x0303 and application must send supported versions in an extra message extension block.

|  |  |  |
| --- | --- | --- |
| Versions | | |
| **Major version** | **Minor version** | **Version type** |
| 3 | 0 | SSL 3.0 |
| 3 | 1 | TLS 1.0 |
| 3 | 2 | TLS 1.1 |
| 3 | 3 | TLS 1.2 |
| 3 | 4 | TLS 1.3 |

Length

The length of "protocol message(s)", "MAC" and "padding" fields combined (i.e. *q*−5), not to exceed 214 bytes (16 KiB).

Protocol message(s)

One or more messages identified by the Protocol field. Note that this field may be encrypted depending on the state of the connection.

MAC and padding

A [message authentication code](https://en.wikipedia.org/wiki/Message_authentication_code) computed over the "protocol message(s)" field, with additional key material included. Note that this field may be encrypted, or not included entirely, depending on the state of the connection.

No "MAC" or "padding" fields can be present at end of TLS records before all cipher algorithms and parameters have been negotiated and handshaked and then confirmed by sending a CipherStateChange record (see below) for signalling that these parameters will take effect in all further records sent by the same peer.

**Handshake protocol**

Most messages exchanged during the setup of the TLS session are based on this record, unless an error or warning occurs and needs to be signaled by an Alert protocol record (see below), or the encryption mode of the session is modified by another record (see ChangeCipherSpec protocol below).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **+** | **Byte +0** | **Byte +1** | **Byte +2** | **Byte +3** |
| **Byte 0** | 22 |  | | |
| **Bytes 1..4** | Version | | Length | |
| *(Major)* | *(Minor)* | *(bits 15..8)* | *(bits 7..0)* |
| **Bytes 5..8** | Message type | Handshake message data length | | |
| *(bits 23..16)* | *(bits 15..8)* | *(bits 7..0)* |
| **Bytes 9..(*n*−1)** | Handshake message data | | | |
| **Bytes *n*..(*n*+3)** | Message type | Handshake message data length | | |
| *(bits 23..16)* | *(bits 15..8)* | *(bits 7..0)* |
| **Bytes (*n*+4)..** | Handshake message data | | | |

Message type

This field identifies the handshake message type.

|  |  |
| --- | --- |
| **Message types** | |
| **Code** | **Description** |
| 0 | HelloRequest |
| 1 | ClientHello |
| 2 | ServerHello |
| 4 | NewSessionTicket |
| 8 | EncryptedExtensions (TLS 1.3 only) |
| 11 | Certificate |
| 12 | ServerKeyExchange |
| 13 | CertificateRequest |
| 14 | ServerHelloDone |
| 15 | CertificateVerify |
| 16 | ClientKeyExchange |
| 20 | Finished |

Handshake message data length

This is a 3-byte field indicating the length of the handshake data, not including the header.

Note that multiple handshake messages may be combined within one record.

**Alert protocol**

This record should normally not be sent during normal handshaking or application exchanges. However, this message can be sent at any time during the handshake and up to the closure of the session. If this is used to signal a fatal error, the session will be closed immediately after sending this record, so this record is used to give a reason for this closure. If the alert level is flagged as a warning, the remote can decide to close the session if it decides that the session is not reliable enough for its needs (before doing so, the remote may also send its own signal).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **+** | **Byte +0** | **Byte +1** | **Byte +2** | **Byte +3** |
| **Byte 0** | 21 |  | | |
| **Bytes 1..4** | Version | | Length | |
| *(Major)* | *(Minor)* | 0 | 2 |
| **Bytes 5..6** | Level | Description |  | |
| **Bytes *7*..(*p*−1)** | [MAC](https://en.wikipedia.org/wiki/Message_authentication_code) (optional) | | | |
| **Bytes *p*..(*q*−1)** | Padding (block ciphers only) | | | |

Level

This field identifies the level of alert. If the level is fatal, the sender should close the session immediately. Otherwise, the recipient may decide to terminate the session itself, by sending its own fatal alert and closing the session itself immediately after sending it. The use of Alert records is optional, however if it is missing before the session closure, the session may be resumed automatically (with its handshakes).

Normal closure of a session after termination of the transported application should preferably be alerted with at least the *Close notify* Alert type (with a simple warning level) to prevent such automatic resume of a new session. Signalling explicitly the normal closure of a secure session before effectively closing its transport layer is useful to prevent or detect attacks (like attempts to truncate the securely transported data, if it intrinsically does not have a predetermined length or duration that the recipient of the secured data may expect).

|  |  |  |
| --- | --- | --- |
| Alert level types | | |
| **Code** | **Level type** | **Connection state** |
| 1 | **warning** | connection or security may be unstable. |
| 2 | **fatal** | connection or security may be compromised, or an unrecoverable error has occurred. |

Description

This field identifies which type of alert is being sent.

|  |  |  |  |
| --- | --- | --- | --- |
| Alert description types | | | |
| **Code** | **Description** | **Level types** | **Note** |
| 0 | Close notify | **warning**/**fatal** |  |
| 10 | Unexpected message | **fatal** |  |
| 20 | Bad record MAC | **fatal** | Possibly a bad SSL implementation, or payload has been tampered with e.g. FTP firewall rule on FTPS server. |
| 21 | Decryption failed | **fatal** | TLS only, reserved |
| 22 | Record overflow | **fatal** | TLS only |
| 30 | Decompression failure | **fatal** |  |
| 40 | Handshake failure | **fatal** |  |
| 41 | No certificate | **warning**/**fatal** | SSL 3.0 only, reserved |
| 42 | Bad certificate | **warning**/**fatal** |  |
| 43 | Unsupported certificate | **warning**/**fatal** | e.g. certificate has only Server authentication usage enabled and is presented as a client certificate |
| 44 | Certificate revoked | **warning**/**fatal** |  |
| 45 | Certificate expired | **warning**/**fatal** | Check server certificate expire also check no certificate in the chain presented has expired |
| 46 | Certificate unknown | **warning**/**fatal** |  |
| 47 | Illegal parameter | **fatal** |  |
| 48 | Unknown CA ([Certificate authority](https://en.wikipedia.org/wiki/Certificate_authority)) | **fatal** | TLS only |
| 49 | Access denied | **fatal** | TLS only – e.g. no client certificate has been presented (TLS: Blank certificate message or SSLv3: No Certificate alert), but server is configured to require one. |
| 50 | Decode error | **fatal** | TLS only |
| 51 | Decrypt error | **warning**/**fatal** | TLS only |
| 60 | Export restriction | **fatal** | TLS only, reserved |
| 70 | Protocol version | **fatal** | TLS only |
| 71 | Insufficient security | **fatal** | TLS only |
| 80 | Internal error | **fatal** | TLS only |
| 86 | Inappropriate Fallback | **fatal** | TLS only |
| 90 | User canceled | **fatal** | TLS only |
| 100 | No renegotiation | **warning** | TLS only |
| 110 | Unsupported extension | **warning** | TLS only |
| 111 | Certificate unobtainable | **warning** | TLS only |
| 112 | Unrecognized name | **warning**/**fatal** | TLS only; client's [Server Name Indicator](https://en.wikipedia.org/wiki/Server_Name_Indication) specified a hostname not supported by the server |
| 113 | Bad certificate status response | **fatal** | TLS only |
| 114 | Bad certificate hash value | **fatal** | TLS only |
| 115 | Unknown [PSK](https://en.wikipedia.org/wiki/Pre-shared_key) identity (used in [TLS-PSK](https://en.wikipedia.org/wiki/TLS-PSK) and [TLS-SRP](https://en.wikipedia.org/wiki/TLS-SRP)) | **fatal** | TLS only |
| 120 | No Application Protocol | **fatal** | TLS only, client's ALPN did not contain any server-supported protocols |

**ChangeCipherSpec protocol**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **+** | **Byte +0** | **Byte +1** | **Byte +2** | **Byte +3** |
| **Byte 0** | 20 |  | | |
| **Bytes 1..4** | Version | | Length | |
| *(Major)* | *(Minor)* | 0 | 1 |
| **Byte 5** | CCS protocol type |  | | |

CCS protocol type

Currently only 1.

**Application protocol**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **+** | **Byte +0** | **Byte +1** | **Byte +2** | **Byte +3** |
| **Byte 0** | 23 |  | | |
| **Bytes 1..4** | Version | | Length | |
| *(Major)* | *(Minor)* | *(bits 15..8)* | *(bits 7..0)* |
| **Bytes 5..(*m*−1)** | Application data | | | |
| **Bytes *m*..(*p*−1)** | [MAC](https://en.wikipedia.org/wiki/Message_authentication_code) (optional) | | | |
| **Bytes *p*..(*q*−1)** | Padding (block ciphers only) | | | |

Length

Length of application data (excluding the protocol header and including the MAC and padding trailers)

MAC

20 bytes for the [SHA-1](https://en.wikipedia.org/wiki/SHA-1)-based [HMAC](https://en.wikipedia.org/wiki/HMAC), 16 bytes for the [MD5](https://en.wikipedia.org/wiki/MD5)-based HMAC.

Padding

Variable length; last byte contains the padding length.

**Support for name-based virtual servers**

From the application protocol point of view, TLS belongs to a lower layer, although the TCP/IP model is too coarse to show it. This means that the TLS handshake is usually (except in the [STARTTLS](https://en.wikipedia.org/wiki/STARTTLS) case) performed before the application protocol can start. In the [name-based virtual server](https://en.wikipedia.org/wiki/Virtual_hosting#Name-based) feature being provided by the application layer, all co-hosted virtual servers share the same certificate because the server has to select and send a certificate immediately after the ClientHello message. This is a big problem in hosting environments because it means either sharing the same certificate among all customers or using a different IP address for each of them.

There are two known workarounds provided by [X.509](https://en.wikipedia.org/wiki/X.509):

* If all virtual servers belong to the same domain, a [wildcard certificate](https://en.wikipedia.org/wiki/Wildcard_certificate) can be used.[[300]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-353) Besides the loose host name selection that might be a problem or not, there is no common agreement about how to match wildcard certificates. Different rules are applied depending on the application protocol or software used.[[301]](https://en.wikipedia.org/wiki/Transport_Layer_Security#cite_note-354)
* Add every virtual host name in the subjectAltName extension. The major problem being that the certificate needs to be reissued whenever a new virtual server is added.

To provide the server name, [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4366](https://tools.ietf.org/html/rfc4366) Transport Layer Security (TLS) Extensions allow clients to include a [Server Name Indication](https://en.wikipedia.org/wiki/Server_Name_Indication) extension (SNI) in the extended ClientHello message. This extension hints to the server immediately which name the client wishes to connect to, so the server can select the appropriate certificate to send to the clients.

[RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [2817](https://tools.ietf.org/html/rfc2817) also documents a method to implement name-based virtual hosting by upgrading HTTP to TLS via an [HTTP/1.1 Upgrade header](https://en.wikipedia.org/wiki/HTTP/1.1_Upgrade_header). Normally this is to securely implement HTTP over TLS within the main "http" [URI scheme](https://en.wikipedia.org/wiki/URI_scheme) (which avoids forking the URI space and reduces the number of used ports), however, few implementations currently support this.[[*citation needed*](https://en.wikipedia.org/wiki/Wikipedia:Citation_needed)]

**Standards**

**Primary standards**

**The current approved version of TLS is version 1.3, which is specified in:**

* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [8446](https://tools.ietf.org/html/rfc8446): "The Transport Layer Security (TLS) Protocol Version 1.3".

**The current standard replaces these former versions, which are now considered obsolete:**

* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [2246](https://tools.ietf.org/html/rfc2246): "The TLS Protocol Version 1.0".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4346](https://tools.ietf.org/html/rfc4346): "The Transport Layer Security (TLS) Protocol Version 1.1".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5246](https://tools.ietf.org/html/rfc5246): "The Transport Layer Security (TLS) Protocol Version 1.2".

**As well as the never standardized SSL 2.0 and 3.0, which are considered obsolete:**

* [Internet Draft (1995)](https://tools.ietf.org/html/draft-hickman-netscape-ssl-00), SSL Version 2.0
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6101](https://tools.ietf.org/html/rfc6101): "The Secure Sockets Layer (SSL) Protocol Version 3.0".

**Extensions**

Other [RFCs](https://en.wikipedia.org/wiki/Request_for_comments) subsequently extended TLS.

**Extensions to TLS 1.0 include:**

* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [2595](https://tools.ietf.org/html/rfc2595): "Using TLS with IMAP, POP3 and ACAP". Specifies an extension to the IMAP, POP3 and ACAP services that allow the server and client to use transport-layer security to provide private, authenticated communication over the Internet.
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [2712](https://tools.ietf.org/html/rfc2712): "Addition of [Kerberos](https://en.wikipedia.org/wiki/Kerberos_(protocol)) Cipher Suites to Transport Layer Security (TLS)". The 40-bit cipher suites defined in this memo appear only for the purpose of documenting the fact that those cipher suite codes have already been assigned.
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [2817](https://tools.ietf.org/html/rfc2817): "Upgrading to TLS Within HTTP/1.1", explains how to use the [Upgrade mechanism in HTTP/1.1](https://en.wikipedia.org/wiki/HTTP/1.1_Upgrade_header) to initiate Transport Layer Security (TLS) over an existing TCP connection. This allows unsecured and secured HTTP traffic to share the same *well known* port (in this case, http: at 80 rather than https: at 443).
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [2818](https://tools.ietf.org/html/rfc2818): "HTTP Over TLS", distinguishes secured traffic from insecure traffic by the use of a different 'server port'.
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [3207](https://tools.ietf.org/html/rfc3207): "SMTP Service Extension for Secure SMTP over Transport Layer Security". Specifies an extension to the SMTP service that allows an SMTP server and client to use transport-layer security to provide private, authenticated communication over the Internet.
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [3268](https://tools.ietf.org/html/rfc3268): "AES Ciphersuites for TLS". Adds [Advanced Encryption Standard](https://en.wikipedia.org/wiki/Advanced_Encryption_Standard) (AES) cipher suites to the previously existing symmetric ciphers.
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [3546](https://tools.ietf.org/html/rfc3546): "Transport Layer Security (TLS) Extensions", adds a mechanism for negotiating protocol extensions during session initialisation and defines some extensions. Made obsolete by [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4366](https://tools.ietf.org/html/rfc4366).
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [3749](https://tools.ietf.org/html/rfc3749): "Transport Layer Security Protocol Compression Methods", specifies the framework for compression methods and the [DEFLATE](https://en.wikipedia.org/wiki/DEFLATE) compression method.
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [3943](https://tools.ietf.org/html/rfc3943): "Transport Layer Security (TLS) Protocol Compression Using Lempel-Ziv-Stac (LZS)".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4132](https://tools.ietf.org/html/rfc4132): "Addition of [Camellia](https://en.wikipedia.org/wiki/Camellia_(cipher)) Cipher Suites to Transport Layer Security (TLS)".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4162](https://tools.ietf.org/html/rfc4162): "Addition of [SEED](https://en.wikipedia.org/wiki/SEED) Cipher Suites to Transport Layer Security (TLS)".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4217](https://tools.ietf.org/html/rfc4217): "Securing [FTP with TLS](https://en.wikipedia.org/wiki/FTPS)".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4279](https://tools.ietf.org/html/rfc4279): "Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)", adds three sets of new cipher suites for the TLS protocol to support authentication based on pre-shared keys.

**Extensions to TLS 1.1 include:**

* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4347](https://tools.ietf.org/html/rfc4347): "[Datagram Transport Layer Security](https://en.wikipedia.org/wiki/Datagram_Transport_Layer_Security)" specifies a TLS variant that works over datagram protocols (such as UDP).
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4366](https://tools.ietf.org/html/rfc4366): "Transport Layer Security (TLS) Extensions" describes both a set of specific extensions and a generic extension mechanism.
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4492](https://tools.ietf.org/html/rfc4492): "[Elliptic Curve Cryptography](https://en.wikipedia.org/wiki/Elliptic_Curve_Cryptography) (ECC) Cipher Suites for Transport Layer Security (TLS)".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4680](https://tools.ietf.org/html/rfc4680): "TLS Handshake Message for Supplemental Data".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4681](https://tools.ietf.org/html/rfc4681): "TLS User Mapping Extension".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4785](https://tools.ietf.org/html/rfc4785): "Pre-Shared Key (PSK) Ciphersuites with NULL Encryption for Transport Layer Security (TLS)".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5054](https://tools.ietf.org/html/rfc5054): "Using the [Secure Remote Password](https://en.wikipedia.org/wiki/Secure_remote_password_protocol) (SRP) Protocol for TLS Authentication". Defines the [TLS-SRP](https://en.wikipedia.org/wiki/TLS-SRP) ciphersuites.
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5077](https://tools.ietf.org/html/rfc5077): "Transport Layer Security (TLS) Session Resumption without Server-Side State".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5081](https://tools.ietf.org/html/rfc5081): "Using [OpenPGP](https://en.wikipedia.org/wiki/OpenPGP) Keys for Transport Layer Security (TLS) Authentication", obsoleted by [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6091](https://tools.ietf.org/html/rfc6091).

**Extensions to TLS 1.2 include:**

* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5288](https://tools.ietf.org/html/rfc5288): "AES [Galois Counter Mode](https://en.wikipedia.org/wiki/Galois/Counter_Mode) (GCM) Cipher Suites for TLS".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5289](https://tools.ietf.org/html/rfc5289): "TLS Elliptic Curve Cipher Suites with SHA-256/384 and AES Galois Counter Mode (GCM)".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5746](https://tools.ietf.org/html/rfc5746): "Transport Layer Security (TLS) Renegotiation Indication Extension".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5878](https://tools.ietf.org/html/rfc5878): "Transport Layer Security (TLS) Authorization Extensions".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5932](https://tools.ietf.org/html/rfc5932): "Camellia Cipher Suites for TLS"
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6066](https://tools.ietf.org/html/rfc6066): "Transport Layer Security (TLS) Extensions: Extension Definitions", includes [Server Name Indication](https://en.wikipedia.org/wiki/Server_Name_Indication) and [OCSP stapling](https://en.wikipedia.org/wiki/OCSP_stapling).
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6091](https://tools.ietf.org/html/rfc6091): "Using [OpenPGP](https://en.wikipedia.org/wiki/OpenPGP) Keys for Transport Layer Security (TLS) Authentication".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6176](https://tools.ietf.org/html/rfc6176): "Prohibiting Secure Sockets Layer (SSL) Version 2.0".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6209](https://tools.ietf.org/html/rfc6209): "Addition of the [ARIA](https://en.wikipedia.org/wiki/ARIA_(cipher)) Cipher Suites to Transport Layer Security (TLS)".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6347](https://tools.ietf.org/html/rfc6347): "Datagram Transport Layer Security Version 1.2".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6367](https://tools.ietf.org/html/rfc6367): "Addition of the Camellia Cipher Suites to Transport Layer Security (TLS)".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6460](https://tools.ietf.org/html/rfc6460): "Suite B Profile for Transport Layer Security (TLS)".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6655](https://tools.ietf.org/html/rfc6655): "AES-CCM Cipher Suites for Transport Layer Security (TLS)".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7027](https://tools.ietf.org/html/rfc7027): "Elliptic Curve Cryptography (ECC) Brainpool Curves for Transport Layer Security (TLS)".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7251](https://tools.ietf.org/html/rfc7251): "AES-CCM Elliptic Curve Cryptography (ECC) Cipher Suites for TLS".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7301](https://tools.ietf.org/html/rfc7301): "Transport Layer Security (TLS) [Application-Layer Protocol Negotiation](https://en.wikipedia.org/wiki/Application-Layer_Protocol_Negotiation) Extension".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7366](https://tools.ietf.org/html/rfc7366): "Encrypt-then-MAC for Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7465](https://tools.ietf.org/html/rfc7465): "Prohibiting RC4 Cipher Suites".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7507](https://tools.ietf.org/html/rfc7507): "TLS Fallback Signaling Cipher Suite Value (SCSV) for Preventing Protocol Downgrade Attacks".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7568](https://tools.ietf.org/html/rfc7568): "Deprecating Secure Sockets Layer Version 3.0".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7627](https://tools.ietf.org/html/rfc7627): "Transport Layer Security (TLS) Session Hash and Extended Master Secret Extension".
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7685](https://tools.ietf.org/html/rfc7685): "A Transport Layer Security (TLS) ClientHello Padding Extension".

**Encapsulations of TLS include:**

* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5216](https://tools.ietf.org/html/rfc5216): "The [EAP](https://en.wikipedia.org/wiki/Extensible_Authentication_Protocol)-TLS Authentication Protocol"

**Informational RFCs**

* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7457](https://tools.ietf.org/html/rfc7457): "Summarizing Known Attacks on Transport Layer Security (TLS) and Datagram TLS (DTLS)"
* [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7525](https://tools.ietf.org/html/rfc7525): "Recommendations for Secure Use of Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)"

**See also**

* [Application-Layer Protocol Negotiation](https://en.wikipedia.org/wiki/Application-Layer_Protocol_Negotiation) – a TLS extension used for SPDY and TLS False Start
* [Bullrun (decryption program)](https://en.wikipedia.org/wiki/Bullrun_(decryption_program)) – a secret anti-encryption program run by the U.S. National Security Agency
* [Certificate authority](https://en.wikipedia.org/wiki/Certificate_authority)
* [Certificate Transparency](https://en.wikipedia.org/wiki/Certificate_Transparency)
* [HTTP Strict Transport Security](https://en.wikipedia.org/wiki/HTTP_Strict_Transport_Security) – HSTS
* [Key ring file](https://en.wikipedia.org/wiki/Key_ring_file)
* [QUIC](https://en.wikipedia.org/wiki/QUIC) (Quick UDP Internet Connections) – "...was designed to provide security protection equivalent to TLS/SSL"; QUIC's main goal is to improve perceived performance of connection-oriented web applications that are currently using TCP
* [Server-Gated Cryptography](https://en.wikipedia.org/wiki/Server-Gated_Cryptography)
* [tcpcrypt](https://en.wikipedia.org/wiki/Tcpcrypt)
* [DTLS](https://en.wikipedia.org/wiki/DTLS)
* [TLS acceleration](https://en.wikipedia.org/wiki/TLS_acceleration)

**References**

 *R. Barnes; M. Thomson; A. Pironti; A. Langley (June 2015).* [*"Deprecating Secure Sockets Layer Version 3.0"*](https://tools.ietf.org/html/rfc7568)*.* [*Archived*](https://web.archive.org/web/20180328080443/https:/tools.ietf.org/html/rfc7568) *from the original on 2018-03-28.*

  *T. Dierks; E. Rescorla (August 2008).* [*"The Transport Layer Security (TLS) Protocol, Version 1.2"*](http://tools.ietf.org/html/rfc5246)*.* [*Archived*](https://web.archive.org/web/20171224222709/https:/tools.ietf.org/html/rfc5246) *from the original on 2017-12-24.*

  [SSL: Intercepted today, decrypted tomorrow](http://news.netcraft.com/archives/2013/06/25/ssl-intercepted-today-decrypted-tomorrow.html) [Archived](https://web.archive.org/web/20130921053551/http:/news.netcraft.com/archives/2013/06/25/ssl-intercepted-today-decrypted-tomorrow.html) 2013-09-21 at the [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine), Netcraft, 2013-06-25.

  *Gothard, Peter.* [*"Google updates SSL certificates to 2048-bit encryption"*](http://www.computing.co.uk/ctg/news/2285984/google-updates-ssl-certificates-to-2048bit-encryption)*. Computing. Incisive Media.* [*Archived*](https://web.archive.org/web/20130922082322/http:/www.computing.co.uk/ctg/news/2285984/google-updates-ssl-certificates-to-2048bit-encryption) *from the original on 22 September 2013. Retrieved 9 September 2013.*

  *A. Freier; P. Karlton; P. Kocher (August 2011).* [*"The Secure Sockets Layer (SSL) Protocol Version 3.0"*](http://tools.ietf.org/html/rfc6101)*.* [*Archived*](https://web.archive.org/web/20120115032035/http:/tools.ietf.org/html/rfc6101) *from the original on 2012-01-15.*

  *Lawrence, Scott; Khare, Rohit.* [*"Upgrading to TLS Within HTTP/1.1"*](https://tools.ietf.org/html/rfc2817)*. tools.ietf.org. Retrieved 15 December 2018.*

  "[SSL/TLS in Detail](https://technet.microsoft.com/en-us/library/cc785811.aspx) [Archived](https://web.archive.org/web/20150206032944/https:/technet.microsoft.com/en-us/library/cc785811.aspx) 2015-02-06 at the [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine)". [*Microsoft TechNet*](https://en.wikipedia.org/wiki/Microsoft_TechNet). Updated July 30, 2003.

  *Hooper, Howard (2012).* [*CCNP Security VPN 642-648 Official Cert Guide*](https://books.google.com/books?id=5PJisOKJ0k8C&pg=PA22) *(2 ed.). Cisco Press. p. 22.* [*ISBN*](https://en.wikipedia.org/wiki/ISBN_(identifier))[*9780132966382*](https://en.wikipedia.org/wiki/Special:BookSources/9780132966382)*.*

  *Spott, Andrew; Leek, Tom; et al.* [*"What layer is TLS?"*](https://security.stackexchange.com/a/93338)*. Information Security Stack Exchange.*

  *T. Dierks, E. Rescorla (August 2008).* [*"Introduction"*](https://tools.ietf.org/html/rfc5246#section-1)*.* [*The Transport Layer Security (TLS) Protocol Version 1.2*](https://tools.ietf.org/html/rfc5246)*. sec. 1.* [*doi*](https://en.wikipedia.org/wiki/Doi_(identifier))*:*[*10.17487/RFC5246*](https://doi.org/10.17487%2FRFC5246)*.* [*RFC*](https://en.wikipedia.org/wiki/RFC_(identifier))[*5246*](https://tools.ietf.org/html/rfc5246)*.*

  *Bright, Peter (17 October 2018).* [*"Apple, Google, Microsoft, and Mozilla come together to end TLS 1.0"*](https://arstechnica.com/gadgets/2018/10/browser-vendors-unite-to-end-support-for-20-year-old-tls-1-0/)*. Retrieved 17 October 2018.*

  [*"Here is what is new and changed in Firefox 74.0 Stable - gHacks Tech News"*](https://www.ghacks.net/2020/03/10/here-is-what-is-new-and-changed-in-firefox-74-0-stable/)*. www.ghacks.net. Retrieved 2020-03-10.*

  [*"TLS 1.0 and TLS 1.1 - Chrome Platform Status"*](https://chromestatus.com/feature/5759116003770368)*. chromestatus.com. Retrieved 2020-03-10.*

  <https://www.circleid.com/posts/20190124_creating_tls_the_pioneering_role_of_ruth_nelson/>

  Thomas Y. C. Woo, Raghuram Bindignavle, Shaowen Su and [Simon S. Lam](https://en.wikipedia.org/wiki/Simon_S._Lam), [*SNP: An interface for secure network programming*](https://www.cs.utexas.edu/users/lam/Vita/Cpapers/WBSL94.pdf) Proceedings USENIX Summer Technical Conference, June **1994**

  *Messmer, Ellen.* [*"Father of SSL, Dr. Taher Elgamal, Finds Fast-Moving IT Projects in the Middle East"*](https://web.archive.org/web/20140531105537/http:/www.networkworld.com/news/2012/120412-elgamal-264739.html)*. Network World. Archived from* [*the original*](http://www.networkworld.com/news/2012/120412-elgamal-264739.html) *on 31 May 2014. Retrieved 30 May 2014.*

  *Greene, Tim.* [*"Father of SSL says despite attacks, the security linchpin has lots of life left"*](https://web.archive.org/web/20140531105257/http:/www.networkworld.com/news/2011/101111-elgamal-251806.html)*. Network World. Archived from* [*the original*](http://www.networkworld.com/news/2011/101111-elgamal-251806.html) *on 31 May 2014. Retrieved 30 May 2014.*

  *Oppliger, Rolf (2016).* [*"Introduction"*](https://books.google.com/books?id=jm6uDgAAQBAJ&pg=PA15)*. SSL and TLS: Theory and Practice (2nd ed.).* [*Artech House*](https://en.wikipedia.org/wiki/Artech_House)*. p. 13.* [*ISBN*](https://en.wikipedia.org/wiki/ISBN_(identifier))[*978-1-60807-999-5*](https://en.wikipedia.org/wiki/Special:BookSources/978-1-60807-999-5)*. Retrieved 2018-03-01 – via Google Books.*

  [*"THE SSL PROTOCOL"*](https://web.archive.org/web/19970614020952/http:/home.netscape.com/newsref/std/SSL.html)*. Netscape Corporation. 2007. Archived from* [*the original*](http://home.netscape.com/newsref/std/SSL.html) *on 14 June 1997.*

  Rescorla 2001

  [*"POODLE: SSLv3 vulnerability (CVE-2014-3566)"*](https://access.redhat.com/articles/1232123)*.* [*Archived*](https://web.archive.org/web/20141205124712/https:/access.redhat.com/articles/1232123) *from the original on 5 December 2014. Retrieved 21 October 2014.*

  [*"Security Standards and Name Changes in the Browser Wars"*](http://tim.dierks.org/2014/05/security-standards-and-name-changes-in.html)*. Retrieved 2020-02-29.*

  *Polk, Tim; McKay, Terry; Chokhani, Santosh (April 2014).* [*"Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations"*](https://web.archive.org/web/20140508025330/http:/nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-52r1.pdf) *(PDF). National Institute of Standards and Technology. p. 67. Archived from* [*the original*](http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-52r1.pdf) *(PDF) on 2014-05-08. Retrieved 2014-05-07.*

  *Laura K. Gray (2015-12-18).* [*"Date Change for Migrating from SSL and Early TLS"*](https://blog.pcisecuritystandards.org/migrating-from-ssl-and-early-tls)*.* [*Payment Card Industry Security Standards Council*](https://en.wikipedia.org/wiki/Payment_Card_Industry_Security_Standards_Council) *blog. Retrieved 2018-04-05.*

  *Company, Newtek - Your Business Solutions.* [*"Changes to PCI Compliance are Coming June 30. Is Your Ecommerce Business Ready?"*](https://www.forbes.com/sites/thesba/2018/05/30/changes-to-pci-compliance-are-coming-june-30-is-your-ecommerce-business-ready/)*. Forbes. Retrieved 2018-06-20.*

  *Dierks, T. & E. Rescorla (April 2006).* [*"The Transport Layer Security (TLS) Protocol Version 1.1"*](http://tools.ietf.org/html/rfc5246#ref-TLS1.1)*.* [*RFC*](https://en.wikipedia.org/wiki/RFC_(identifier))[*4346*](https://tools.ietf.org/html/rfc4346)*.* [*Archived*](https://web.archive.org/web/20171224222709/https:/tools.ietf.org/html/rfc5246#ref-TLS1.1) *from the original on 2017-12-24.*

  *T. Dierks, E. Rescorla (August 2008).* [*"Finished"*](https://tools.ietf.org/html/rfc5246#section-7.4.9)*.* [*The Transport Layer Security (TLS) Protocol Version 1.2*](https://tools.ietf.org/html/rfc5246)*. sec. 7.4.9.* [*doi*](https://en.wikipedia.org/wiki/Doi_(identifier))*:*[*10.17487/RFC5246*](https://doi.org/10.17487%2FRFC5246)*.* [*RFC*](https://en.wikipedia.org/wiki/RFC_(identifier))[*5246*](https://tools.ietf.org/html/rfc5246)*.*

  [*"Differences between TLS 1.2 and TLS 1.3 (#TLS13)"*](https://web.archive.org/web/20190919000200/https:/www.wolfssl.com/differences-between-tls-12-and-tls-13-9/)*. WolfSSL. 18 September 2019. Archived from* [*the original*](https://www.wolfssl.com/differences-between-tls-12-and-tls-13-9/) *on 19 September 2019. Retrieved 18 September 2019.*

  [*"NSS 3.29 release notes"*](https://developer.mozilla.org/en-US/docs/Mozilla/Projects/NSS/NSS_3.29_release_notes)*. Mozilla Developer Network. February 2017.* [*Archived*](https://web.archive.org/web/20170222052829/https:/developer.mozilla.org/en-US/docs/Mozilla/Projects/NSS/NSS_3.29_release_notes) *from the original on 2017-02-22.*

  [*"Enable TLS 1.3 by default"*](https://bugzilla.mozilla.org/show_bug.cgi?id=1310516)*. Bugzilla@Mozilla. 16 October 2016. Retrieved 10 October 2017.*

  [*"Firefox — Notes (60.0)"*](https://www.mozilla.org/en-US/firefox/60.0/releasenotes/)*. Mozilla. Retrieved 2018-05-10.*

  [*"ProxySG, ASG and WSS will interrupt SSL connections when clients using TLS 1.3 access sites also using TLS 1.3"*](http://bluecoat.force.com/knowledgebase/articles/Technical_Alert/000032878)*. BlueTouch Online. 16 May 2017.* [*Archived*](https://web.archive.org/web/20170912061432/http:/bluecoat.force.com/knowledgebase/articles/Technical_Alert/000032878) *from the original on 12 September 2017. Retrieved 11 September 2017.*

  [*"TLS 1.3 IETF 100 Hackathon"*](https://web.archive.org/web/20180115220635/https:/datatracker.ietf.org/meeting/100/materials/slides-100-hackathon-sessa-tls-13/)*. Archived from* [*the original*](https://datatracker.ietf.org/meeting/100/materials/slides-100-hackathon-sessa-tls-13/) *on 2018-01-15.*

  *IETF – Internet Engineering Task Force (2017-11-12),* [*IETF Hackathon Presentations and Awards*](https://www.youtube.com/watch?v=33XW5yzjtME&t=2338)*, retrieved 2017-11-14*

  [*"Hurrah! TLS 1.3 is here. Now to implement it and put it into software"*](https://www.theregister.co.uk/2018/03/27/with_tls_13_signed_off_its_implementation_time)*. Retrieved 2018-03-28.*

  *IETF - Internet Engineering Task Force (2018-07-15),* [*IETF102-HACKATHON-20180715-1400*](https://www.youtube.com/watch?v=u6rz4PWA_As&t=4526)*, retrieved 2018-07-18*

  [*"wolfSSL TLS 1.3 BETA Release Now Available"*](https://www.wolfssl.com/wolfssl-tls-1-3-beta-release-now-available/)*. info@wolfssl.com. 11 May 2017. Retrieved 11 May 2017.*

  [*"TLS 1.3 PROTOCOL SUPPORT"*](https://www.wolfssl.com/docs/tls13/)*. info@wolfssl.com.*

  [*"TLS 1.3 Draft 28 Support in wolfSSL"*](https://www.wolfssl.com/tls-1-3-draft-28-support-wolfssl/)*. info@wolfssl.com. 14 June 2018. Retrieved 14 June 2018.*

  [*"OpenSSL 1.1.1 Is Released"*](https://www.openssl.org/blog/blog/2018/09/11/release111/)*. Matt Caswell. 11 Sep 2018. Retrieved 19 Dec 2018.*

  *Hoffman-Andrews, Jacob (2019-02-26).* [*"ETS Isn't TLS and You Shouldn't Use It"*](https://www.eff.org/deeplinks/2019/02/ets-isnt-tls-and-you-shouldnt-use-it)*.* [*Electronic Frontier Foundation*](https://en.wikipedia.org/wiki/Electronic_Frontier_Foundation)*. Retrieved 2019-02-27.*

  *Rea, Scott (2013).* [*"Alternatives to Certification Authorities for a Secure Web"*](https://www.rsaconference.com/writable/presentations/file_upload/sec-t02_final.pdf) *(PDF). RSA Conference Asia Pacific.* [*Archived*](https://web.archive.org/web/20161007222635/https:/www.rsaconference.com/writable/presentations/file_upload/sec-t02_final.pdf) *(PDF) from the original on 7 October 2016. Retrieved 7 September 2016.*

  [Counting SSL certificates; netcraft; May 13, 2015.](http://news.netcraft.com/archives/2015/05/13/counting-ssl-certificates.html) [Archived](https://web.archive.org/web/20150516035536/http:/news.netcraft.com/archives/2015/05/13/counting-ssl-certificates.html) May 16, 2015, at the [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine)

  *Raymond, Art (3 August 2017).* [*"Lehi's DigiCert swallows web security competitor in $1 billion deal"*](https://www.deseretnews.com/article/865686081/Lehis-DigiCert-swallows-web-security-competitor-in-1-billion-deal.html)*. Deseret News. Retrieved 21 May 2020.*

  [*"Market share trends for SSL certificate authorities"*](https://w3techs.com/technologies/history_overview/ssl_certificate)*. W3Techs. Retrieved 21 May 2020.*

  [Law Enforcement Appliance Subverts SSL](https://www.wired.com/threatlevel/2010/03/packet-forensics/) [Archived](https://web.archive.org/web/20140315064727/http:/www.wired.com/threatlevel/2010/03/packet-forensics) 2014-03-15 at the [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine), Wired, 2010-04-03.

  [New Research Suggests That Governments May Fake SSL Certificates](https://www.eff.org/deeplinks/2010/03/researchers-reveal-likelihood-governments-fake-ssl) [Archived](https://web.archive.org/web/20160104234608/https:/www.eff.org/deeplinks/2010/03/researchers-reveal-likelihood-governments-fake-ssl) 2016-01-04 at the [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine), EFF, 2010-03-24.

  *P. Eronen, Ed.* [*"Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)"*](https://tools.ietf.org/html/rfc4279)*. Internet Engineering Task Force.* [*RFC*](https://en.wikipedia.org/wiki/RFC_(identifier))[*4279*](https://tools.ietf.org/html/rfc4279)*.* [*Archived*](https://web.archive.org/web/20130905074722/http:/tools.ietf.org/html/rfc4279) *from the original on 5 September 2013. Retrieved 9 September 2013.*

  *D. Taylor, Ed.* [*"Using the Secure Remote Password (SRP) Protocol for TLS Authentication"*](https://tools.ietf.org/html/rfc5054)*. Internet Engineering Task Force.* [*RFC*](https://en.wikipedia.org/wiki/RFC_(identifier))[*5054*](https://tools.ietf.org/html/rfc5054)*.* [*Archived*](https://web.archive.org/web/20141207151603/http:/tools.ietf.org/html/rfc5054) *from the original on December 7, 2014. Retrieved December 21, 2014.*

  [*"The value of 2,048-bit encryption: Why encryption key length matters"*](http://searchsecurity.techtarget.com/answer/From-1024-to-2048-bit-The-security-effect-of-encryption-key-length)*. SearchSecurity.* [*Archived*](https://web.archive.org/web/20180116081141/http:/searchsecurity.techtarget.com/answer/From-1024-to-2048-bit-The-security-effect-of-encryption-key-length) *from the original on 2018-01-16. Retrieved 2017-12-18.*

  *Sean Turner (September 17, 2015).* [*"Consensus: remove DSA from TLS 1.3"*](https://www.ietf.org/mail-archive/web/tls/current/msg17680.html)*.* [*Archived*](https://web.archive.org/web/20151003193113/http:/www.ietf.org/mail-archive/web/tls/current/msg17680.html) *from the original on October 3, 2015.*

  [RFC 8422](https://tools.ietf.org/html/rfc8422)

  [draft-chudov-cryptopro-cptls-04 – GOST 28147-89 Cipher Suites for Transport Layer Security (TLS)](https://tools.ietf.org/html/draft-chudov-cryptopro-cptls-04)

  [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5288](https://tools.ietf.org/html/rfc5288), [5289](https://tools.ietf.org/html/rfc5289)

  [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6655](https://tools.ietf.org/html/rfc6655), [7251](https://tools.ietf.org/html/rfc7251)

  [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6367](https://tools.ietf.org/html/rfc6367)

  [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5932](https://tools.ietf.org/html/rfc5932), [6367](https://tools.ietf.org/html/rfc6367)

  [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6209](https://tools.ietf.org/html/rfc6209)

  [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [4162](https://tools.ietf.org/html/rfc4162)

  [*"On the Practical (In-)Security of 64-bit Block Ciphers — Collision Attacks on HTTP over TLS and OpenVPN"*](https://sweet32.info/SWEET32_CCS16.pdf) *(PDF). 2016-10-28.* [*Archived*](https://web.archive.org/web/20170424021101/https:/sweet32.info/SWEET32_CCS16.pdf) *(PDF) from the original on 2017-04-24. Retrieved 2017-06-08.*

  [*"NIST Special Publication 800-57 Recommendation for Key Management — Part 1: General (Revised)"*](https://web.archive.org/web/20140606050814/http:/csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1-revised2_Mar08-2007.pdf) *(PDF). 2007-03-08. Archived from* [*the original*](http://csrc.nist.gov/publications/nistpubs/800-57/sp800-57-Part1-revised2_Mar08-2007.pdf) *(PDF) on June 6, 2014. Retrieved 2014-07-03.*

  *Qualys SSL Labs.* [*"SSL/TLS Deployment Best Practices"*](https://www.ssllabs.com/projects/best-practices/index.html)*.* [*Archived*](https://web.archive.org/web/20150704101956/https:/www.ssllabs.com/projects/best-practices/index.html) *from the original on 4 July 2015. Retrieved 2 June 2015.*

  [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5469](https://tools.ietf.org/html/rfc5469)

  [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [7905](https://tools.ietf.org/html/rfc7905)

  [AEAD Ciphers](https://tools.ietf.org/html/rfc5246#section-6.2.3.3)

  [*"Http vs https"*](https://www.instantssl.com/ssl-certificate-products/https.html)*.* [*Archived*](https://web.archive.org/web/20150212105201/https:/www.instantssl.com/ssl-certificate-products/https.html) *from the original on 2015-02-12. Retrieved 2015-02-12.*

  As of August 3, 2019. [*"SSL Pulse: Survey of the SSL Implementation of the Most Popular Websites"*](https://www.ssllabs.com/ssl-pulse/)*.* [*Qualys*](https://en.wikipedia.org/wiki/Qualys)*. Retrieved 2019-09-01.*

  *ivanr.* [*"RC4 in TLS is Broken: Now What?"*](https://community.qualys.com/blogs/securitylabs/2013/03/19/rc4-in-tls-is-broken-now-what)*. Qualsys Security Labs.* [*Archived*](https://web.archive.org/web/20130827044512/https:/community.qualys.com/blogs/securitylabs/2013/03/19/rc4-in-tls-is-broken-now-what) *from the original on 2013-08-27. Retrieved 2013-07-30.*

  *Bodo Möller, Thai Duong & Krzysztof Kotowicz.* [*"This POODLE Bites: Exploiting The SSL 3.0 Fallback"*](https://www.openssl.org/~bodo/ssl-poodle.pdf) *(PDF).* [*Archived*](https://web.archive.org/web/20141014224443/https:/www.openssl.org/~bodo/ssl-poodle.pdf) *(PDF) from the original on 2014-10-14. Retrieved 2014-10-15.*

  [*"What browsers support Extended Validation (EV) and display an EV indicator?"*](https://web.archive.org/web/20151231171309/https:/knowledge.verisign.com/support/ssl-certificates-support/index?page=content&id=SO10090&actp=search&viewlocale=en_US&searchid=1406590748892)*.* [*Symantec*](https://en.wikipedia.org/wiki/NortonLifeLock)*. Archived from* [*the original*](https://knowledge.verisign.com/support/ssl-certificates-support/index?page=content&id=SO10090&actp=search&viewlocale=en_US&searchid=1406590748892) *on 2015-12-31. Retrieved 2014-07-28.*

  [*"SHA-256 Compatibility"*](https://support.globalsign.com/customer/portal/articles/1499561-sha-256-compatibility)*.* [*Archived*](https://web.archive.org/web/20150701004324/https:/support.globalsign.com/customer/portal/articles/1499561-sha-256-compatibility) *from the original on 2015-07-01. Retrieved 2015-06-12.*

  [*"ECC Compatibility"*](https://support.globalsign.com/customer/portal/articles/1995283-ecc-compatibility)*.* [*Archived*](https://web.archive.org/web/20160217122951/https:/support.globalsign.com/customer/portal/articles/1995283-ecc-compatibility) *from the original on 2016-02-17. Retrieved 2015-06-13.*

  [*"Tracking the FREAK Attack"*](https://freakattack.com/)*.* [*Archived*](https://web.archive.org/web/20150306174623/https:/freakattack.com/) *from the original on 2015-03-06. Retrieved 2015-03-08.*

  [*"FREAK: Factoring RSA Export Keys"*](https://www.smacktls.com/#freak)*.* [*Archived*](https://web.archive.org/web/20150311112517/https:/www.smacktls.com/#freak) *from the original on 2015-03-11. Retrieved 2015-03-08.*

  *Google (2012-05-29).* [*"Dev Channel Update"*](http://googlechromereleases.blogspot.nl/2012/05/dev-channel-update_29.html)*.* [*Archived*](https://web.archive.org/web/20130302032339/http:/googlechromereleases.blogspot.nl/2012/05/dev-channel-update_29.html) *from the original on 2013-03-02. Retrieved 2011-06-01.*

  *Google (2012-08-21).* [*"Stable Channel Update"*](https://googlechromereleases.blogspot.co.uk/2012/08/stable-channel-update_21.html)*.* [*Archived*](https://web.archive.org/web/20120825192441/http:/googlechromereleases.blogspot.co.uk/2012/08/stable-channel-update_21.html) *from the original on 2012-08-25. Retrieved 2012-08-22.*

  *Chromium Project (2013-05-30).* [*"Chromium TLS 1.2 Implementation"*](https://src.chromium.org/viewvc/chrome?revision=203090&view=revision)*.*

  [*"The Chromium Project: BoringSSL"*](https://www.chromium.org/Home/chromium-security/boringssl)*.* [*Archived*](https://web.archive.org/web/20150923033459/http:/www.chromium.org/Home/chromium-security/boringssl) *from the original on 2015-09-23. Retrieved 2015-09-05.*

  [*"Chrome Stable Release"*](http://googlechromereleases.blogspot.jp/2011/10/chrome-stable-release.html)*. Chrome Releases. 2011-10-25.* [*Archived*](https://web.archive.org/web/20150220020306/http:/googlechromereleases.blogspot.jp/2011/10/chrome-stable-release.html) *from the original on 2015-02-20. Retrieved 2015-02-01.*

  [*"SVN revision log on Chrome 10.0.648.127 release"*](https://archive.today/20140619142454/http:/build.chromium.org/f/chromium/perf/dashboard/ui/changelog.html?url=/trunk/src&range=72316:67679&mode=html)*. Archived from* [*the original*](https://build.chromium.org/f/chromium/perf/dashboard/ui/changelog.html?url=/trunk/src&range=72316:67679&mode=html) *on 2014-06-19. Retrieved 2014-06-19.*

  [*"ImperialViolet – CRIME"*](https://www.imperialviolet.org/2012/09/21/crime.html)*. 2012-09-22.* [*Archived*](https://web.archive.org/web/20150110195746/https:/www.imperialviolet.org/2012/09/21/crime.html) *from the original on 2015-01-10. Retrieved 2014-10-18.*

  [*"SSL/TLS Overview"*](https://sites.google.com/site/tlsssloverview/ssl-v-tls/tls-versions-and-browser-compatability)*. 2008-08-06.* [*Archived*](https://web.archive.org/web/20130703043525/https:/sites.google.com/site/tlsssloverview/ssl-v-tls/tls-versions-and-browser-compatability) *from the original on 2013-07-03. Retrieved 2013-03-29.*

  [*"Chromium Issue 90392"*](https://code.google.com/p/chromium/issues/detail?id=90392)*. 2008-08-06.* [*Archived*](https://web.archive.org/web/20130803110745/http:/code.google.com/p/chromium/issues/detail?id=90392) *from the original on 2013-08-03. Retrieved 2013-06-28.*

  [*"Issue 23503030 Merge 219882"*](https://codereview.chromium.org/23503030/)*. 2013-09-03.* [*Archived*](https://web.archive.org/web/20140226230020/https:/codereview.chromium.org/23503030/) *from the original on 2014-02-26. Retrieved 2013-09-19.*

  [*"Issue 278370: Unable to submit client certificates over TLS 1.2 from Windows"*](https://code.google.com/p/chromium/issues/detail?id=278370)*. 2013-08-23.* [*Archived*](https://web.archive.org/web/20131005021656/http:/code.google.com/p/chromium/issues/detail?id=278370) *from the original on 2013-10-05. Retrieved 2013-10-03.*

  *Möller, Bodo (2014-10-14).* [*"This POODLE bites: exploiting the SSL 3.0 fallback"*](http://googleonlinesecurity.blogspot.com/2014/10/this-poodle-bites-exploiting-ssl-30.html)*. Google Online Security blog. Google (via Blogspot).* [*Archived*](https://web.archive.org/web/20141028003952/http:/googleonlinesecurity.blogspot.com/2014/10/this-poodle-bites-exploiting-ssl-30.html) *from the original on 2014-10-28. Retrieved 2014-10-28.*

  [*"An update on SSLv3 in Chrome"*](https://groups.google.com/a/chromium.org/forum/#!topic/security-dev/Vnhy9aKM_l4)*. Security-dev. 2014-10-31. Retrieved 2014-11-04.*

  [*"Stable Channel Update"*](http://googlechromereleases.blogspot.com/2014/02/stable-channel-update_20.html)*. Mozilla Developer Network. 2014-02-20.* [*Archived*](https://web.archive.org/web/20141024210105/http:/googlechromereleases.blogspot.com/2014/02/stable-channel-update_20.html) *from the original on 2014-10-24. Retrieved 2014-11-14.*

  [*"Changelog for Chrome 33.0.1750.117"*](https://archive.today/20140116153037/http:/build.chromium.org/f/chromium/perf/dashboard/ui/changelog.html?url=/trunk/src&range=232870:241107&mode=html)*. Google. Archived from* [*the original*](https://build.chromium.org/f/chromium/perf/dashboard/ui/changelog.html?url=/trunk/src&range=232870:241107&mode=html) *on 2014-01-16. Retrieved 2014-11-14.*

  [*"Issue 318442: Update to NSS 3.15.3 and NSPR 4.10.2"*](https://code.google.com/p/chromium/issues/detail?id=318442)*.* [*Archived*](https://web.archive.org/web/20150315043755/https:/code.google.com/p/chromium/issues/detail?id=318442) *from the original on 2015-03-15. Retrieved 2014-11-14.*

  [*"Issue 693963003: Add minimum TLS version control to about:flags and Finch gate it. – Code Review"*](https://codereview.chromium.org/693963003)*.* [*Archived*](https://web.archive.org/web/20150416072943/https:/codereview.chromium.org/693963003) *from the original on 2015-04-16. Retrieved 2015-01-22.*

  [*"Issue 375342: Drop RC4 Support"*](https://code.google.com/p/chromium/issues/detail?id=375342)*.* [*Archived*](https://web.archive.org/web/20150912071452/https:/code.google.com/p/chromium/issues/detail?id=375342) *from the original on 2015-09-12. Retrieved 2015-05-22.*

  [*"Issue 436391: Add info on end of life of SSLVersionFallbackMin & SSLVersionMin policy in documentation"*](https://code.google.com/p/chromium/issues/detail?id=436391)*.* [*Archived*](https://web.archive.org/web/20150418095600/http:/code.google.com/p/chromium/issues/detail?id=436391) *from the original on 2015-04-18. Retrieved 2015-04-19.*

  [*"Issue 490240: Increase minimum DH size to 1024 bits (tracking bug)"*](https://code.google.com/p/chromium/issues/detail?id=490240)*.* [*Archived*](https://web.archive.org/web/20150912081831/https:/code.google.com/p/chromium/issues/detail?id=490240) *from the original on 2015-09-12. Retrieved 2015-05-29.*

  [*"Intent to deprecate: RC4"*](https://groups.google.com/a/chromium.org/forum/#!msg/security-dev/kVfCywocUO8/2BW3INFdDwAJ)*. Retrieved 2015-12-21.*

  [*"An update on SHA-1 certificates in Chrome"*](https://googleonlinesecurity.blogspot.com/2015/12/an-update-on-sha-1-certificates-in.html)*. 2015-12-18.* [*Archived*](https://web.archive.org/web/20151218214756/https:/googleonlinesecurity.blogspot.com/2015/12/an-update-on-sha-1-certificates-in.html) *from the original on 2015-12-18. Retrieved 2015-12-21.*

  <https://docs.microsoft.com/en-us/microsoft-edge/web-platform/site-impacting-changes>

  [*"Security in Firefox 2"*](https://developer.mozilla.org/en-US/Firefox/Releases/2/Security_changes)*. 2008-08-06.* [*Archived*](https://web.archive.org/web/20140714142705/https:/developer.mozilla.org/en-US/Firefox/Releases/2/Security_changes) *from the original on 2014-07-14. Retrieved 2009-03-31.*

  [*"Attack against TLS-protected communications"*](https://blog.mozilla.org/security/2011/09/27/attack-against-tls-protected-communications/)*. Mozilla Security Blog. Mozilla. 2011-09-27.* [*Archived*](https://web.archive.org/web/20150304221307/https:/blog.mozilla.org/security/2011/09/27/attack-against-tls-protected-communications/) *from the original on 2015-03-04. Retrieved 2015-02-01.*

  [*"Introduction to SSL"*](https://developer.mozilla.org/en-US/docs/Introduction_to_SSL)*. MDN.* [*Archived*](https://web.archive.org/web/20140714205945/https:/developer.mozilla.org/en-US/docs/Introduction_to_SSL) *from the original on 2014-07-14. Retrieved 2014-06-19.*

  [*"NSS 3.15.3 Release Notes"*](https://developer.mozilla.org/en-US/docs/Mozilla/Projects/NSS/NSS_3.15.3_release_notes)*. Mozilla Developer Network. Mozilla.* [*Archived*](https://web.archive.org/web/20140605001016/https:/developer.mozilla.org/en-US/docs/Mozilla/Projects/NSS/NSS_3.15.3_release_notes) *from the original on 2014-06-05. Retrieved 2014-07-13.*

  [*"MFSA 2013-103: Miscellaneous Network Security Services (NSS) vulnerabilities"*](https://www.mozilla.org/security/announce/2013/mfsa2013-103.html)*. Mozilla. Mozilla.* [*Archived*](https://web.archive.org/web/20140714121632/http:/www.mozilla.org/security/announce/2013/mfsa2013-103.html) *from the original on 2014-07-14. Retrieved 2014-07-13.*

  [*"Bug 565047 – (RFC4346) Implement TLS 1.1 (RFC 4346)"*](https://bugzilla.mozilla.org/show_bug.cgi?id=565047)*. Retrieved 2013-10-29.*

  [*"Bug 480514 – Implement support for TLS 1.2 (RFC 5246)"*](https://bugzilla.mozilla.org/show_bug.cgi?id=480514)*. Retrieved 2013-10-29.*

  [*"Bug 733647 – Implement TLS 1.1 (RFC 4346) in Gecko (Firefox, Thunderbird), on by default"*](https://bugzilla.mozilla.org/show_bug.cgi?id=733647)*. Retrieved 2013-12-04.*

  [*"Firefox Notes – Desktop"*](https://website-archive.mozilla.org/www.mozilla.org/firefox_releasenotes/en-US/firefox/27.0/releasenotes/)*. 2014-02-04.* [*Archived*](https://web.archive.org/web/20140207095334/https:/website-archive.mozilla.org/www.mozilla.org/firefox_releasenotes/en-US/firefox/27.0/releasenotes/) *from the original on 2014-02-07. Retrieved 2014-02-04.*

  [*"Bug 861266 – Implement TLS 1.2 (RFC 5246) in Gecko (Firefox, Thunderbird), on by default"*](https://bugzilla.mozilla.org/show_bug.cgi?id=861266)*. Retrieved 2013-11-18.*

  [*"The POODLE Attack and the End of SSL 3.0"*](https://blog.mozilla.org/security/2014/10/14/the-poodle-attack-and-the-end-of-ssl-3-0/)*. Mozilla blog. Mozilla. 2014-10-14.* [*Archived*](https://web.archive.org/web/20141018231351/https:/blog.mozilla.org/security/2014/10/14/the-poodle-attack-and-the-end-of-ssl-3-0/) *from the original on 2014-10-18. Retrieved 2014-10-28.*

  [*"Firefox — Notes (34.0) — Mozilla"*](https://www.mozilla.org/en-US/firefox/34.0/releasenotes/)*. mozilla.org. 2014-12-01.* [*Archived*](https://web.archive.org/web/20150409022558/https:/www.mozilla.org/en-US/firefox/34.0/releasenotes/) *from the original on 2015-04-09. Retrieved 2015-04-03.*

  [*"Bug 1083058 – A pref to control TLS version fallback"*](https://bugzilla.mozilla.org/show_bug.cgi?id=1083058)*. bugzilla.mozilla.org. Retrieved 2014-11-06.*

  [*"Bug 1036737 – Add support for draft-ietf-tls-downgrade-scsv to Gecko/Firefox"*](https://bugzilla.mozilla.org/show_bug.cgi?id=1036737)*. bugzilla.mozilla.org. Retrieved 2014-10-29.*

  [*"Bug 1166031 – Update to NSS 3.19.1"*](https://bugzilla.mozilla.org/show_bug.cgi?id=1166031)*. bugzilla.mozilla.org. Retrieved 2015-05-29.*

  [*"Bug 1088915 – Stop offering RC4 in the first handshakes"*](https://bugzilla.mozilla.org/show_bug.cgi?id=1088915)*. bugzilla.mozilla.org. Retrieved 2014-11-04.*

  [*"Firefox — Notes (39.0) — Mozilla"*](https://www.mozilla.org/en-US/firefox/39.0/releasenotes/)*. mozilla.org. 2015-06-30.* [*Archived*](https://web.archive.org/web/20150703163249/https:/www.mozilla.org/en-US/firefox/39.0/releasenotes/) *from the original on 2015-07-03. Retrieved 2015-07-03.*

  [*"Google, Microsoft, and Mozilla will drop RC4 encryption in Chrome, Edge, IE, and Firefox next year"*](https://venturebeat.com/2015/09/01/google-microsoft-and-mozilla-will-drop-rc4-support-in-chrome-edge-ie-and-firefox-next-year/)*. VentureBeat. 2015-09-01.* [*Archived*](https://web.archive.org/web/20150905214154/http:/venturebeat.com/2015/09/01/google-microsoft-and-mozilla-will-drop-rc4-support-in-chrome-edge-ie-and-firefox-next-year/) *from the original on 2015-09-05. Retrieved 2015-09-05.*

  [*"Intent to ship: RC4 disabled by default in Firefox 44"*](https://groups.google.com/forum/#!searchin/mozilla.dev.platform/rc4/mozilla.dev.platform/JIEFcrGhqSM/CIjtpwxoLQAJ)*.* [*Archived*](http://arquivo.pt/wayback/20110122130054/https:/groups.google.com/forum/#!searchin/mozilla.dev.platform/rc4/mozilla.dev.platform/JIEFcrGhqSM/CIjtpwxoLQAJ) *from the original on 2011-01-22. Retrieved 2015-10-18.*

  [*"RC4 is now allowed only on whitelisted sites (Reverted)"*](https://www.fxsitecompat.com/en-US/docs/2015/rc4-is-now-allowed-only-on-whitelisted-sites/)*. Retrieved 2015-11-02.*

  [*"Firefox — Notes (44.0) — Mozilla"*](https://www.mozilla.org/en-US/firefox/44.0/releasenotes/)*. mozilla.org. 2016-01-26.* [*Archived*](https://web.archive.org/web/20160304025010/https:/www.mozilla.org/en-US/firefox/44.0/releasenotes/) *from the original on 2016-03-04. Retrieved 2016-03-09.*

  [*"Bug 1342082 – Disable TLS 1.3 for FF52 Release"*](https://bugzilla.mozilla.org/show_bug.cgi?id=1342082)*. Retrieved 2017-03-29.*

  <https://www.mozilla.org/en-US/firefox/78.0/releasenotes/>

  [*"Opera 9.0 for Windows Changelog"*](http://www.opera.com/docs/changelogs/windows/900/)*.* [*Archived*](https://archive.today/20120910/http:/www.opera.com/docs/changelogs/windows/900/) *from the original on 2012-09-10.*

  [*"Opera 2 series"*](http://www.opera.com/docs/history/presto/#o2)*.* [*Archived*](https://web.archive.org/web/20141023073434/http:/www.opera.com/docs/history/presto/#o2) *from the original on 2014-10-23. Retrieved 2014-09-20.*

  [*"Opera 3 series"*](http://www.opera.com/docs/history/presto/#o3)*.* [*Archived*](https://web.archive.org/web/20141023073434/http:/www.opera.com/docs/history/presto/#o3) *from the original on 2014-10-23. Retrieved 2014-09-20.*

  [*"Opera 4 series"*](http://www.opera.com/docs/history/presto/#o4)*.* [*Archived*](https://web.archive.org/web/20141023073434/http:/www.opera.com/docs/history/presto/#o4) *from the original on 2014-10-23. Retrieved 2014-09-20.*

  [*"Changelog for Opera 5.x for Windows"*](http://www.opera.com/docs/changelogs/windows/500-512/)*.* [*Archived*](https://web.archive.org/web/20141019005943/http:/www.opera.com/docs/changelogs/windows/500-512/) *from the original on 2014-10-19. Retrieved 2014-06-19.*

  [*"Changelog for Opera [8] Beta 2 for Windows"*](http://www.opera.com/docs/changelogs/windows/800b2/)*.* [*Archived*](https://web.archive.org/web/20051123055608/http:/opera.com/docs/changelogs/windows/800b2/) *from the original on 2005-11-23. Retrieved 2014-06-19.*

  [*"Web Specifications Supported in Opera 9"*](http://www.opera.com/docs/specs/opera9/)*.* [*Archived*](https://web.archive.org/web/20141026185804/http:/www.opera.com/docs/specs/opera9/) *from the original on 2014-10-26. Retrieved 2014-06-19.*

  [*"Opera: Opera 10 beta for Windows changelog"*](https://web.archive.org/web/20141023083043/http:/www.opera.com/docs/changelogs/windows/1000b1/)*. Archived from* [*the original*](http://www.opera.com/docs/changelogs/windows/1000b1/) *on 2014-10-23. Retrieved 2014-06-19.*

  [*"About Opera 11.60 and new problems with some secure servers"*](https://web.archive.org/web/20120118122036/http:/my.opera.com/securitygroup/blog/2011/12/11/opera-11-60-and-new-problems-with-some-secure-servers)*. 2011-12-11. Archived from* [*the original*](http://my.opera.com/securitygroup/blog/2011/12/11/opera-11-60-and-new-problems-with-some-secure-servers) *on 2012-01-18.*

  [*"Security changes in Opera 25; the poodle attacks"*](http://blogs.opera.com/security/2014/10/security-changes-opera-25-poodle-attacks/)*. 2014-10-15.* [*Archived*](https://web.archive.org/web/20141020091840/http:/blogs.opera.com/security/2014/10/security-changes-opera-25-poodle-attacks/) *from the original on 2014-10-20. Retrieved 2014-10-28.*

  [*"Unjam the logjam"*](http://blogs.opera.com/security/2015/06/unjam-the-logjam/)*. 2015-06-09.* [*Archived*](https://web.archive.org/web/20150614005330/http:/blogs.opera.com/security/2015/06/unjam-the-logjam/) *from the original on 2015-06-14. Retrieved 2015-06-11.*

  [*"Advisory: RC4 encryption protocol is vulnerable to certain brute force attacks"*](http://www.opera.com/ja/security/advisory/1046)*. 2013-04-04.* [*Archived*](https://web.archive.org/web/20150315052403/http:/www.opera.com/ja/security/advisory/1046) *from the original on 2015-03-15. Retrieved 2014-11-14.*

  [*"On the Precariousness of RC4"*](https://web.archive.org/web/20131112235240/http:/my.opera.com/securitygroup/blog/2013/03/20/on-the-precariousness-of-rc4)*. 2013-03-20. Archived from* [*the original*](http://my.opera.com/securitygroup/blog/2013/03/20/on-the-precariousness-of-rc4) *on 2013-11-12. Retrieved 2014-11-17.*

  [*"Opera 12 and Opera Mail security update"*](http://www.opera.com/blogs/security/2016/02/opera-12-and-opera-mail-security-update/)*. 2016-02-16.* [*Archived*](https://web.archive.org/web/20160216143015/http:/www.opera.com/blogs/security/2016/02/opera-12-and-opera-mail-security-update/) *from the original on 2016-02-16. Retrieved 2016-02-17.*

  [*"Dev.Opera — Opera 14 for Android Is Out!"*](https://dev.opera.com/blog/opera-14-for-android-is-out/)*. 2013-05-21.* [*Archived*](https://web.archive.org/web/20150130172538/https:/dev.opera.com/blog/opera-14-for-android-is-out/) *from the original on 2015-01-30. Retrieved 2014-09-23.*

  [*"Dev.Opera — Introducing Opera 15 for Computers, and a Fast Release Cycle"*](https://dev.opera.com/blog/introducing-opera-15-for-desktop-and-a-fast-release-cycle/)*. 2013-07-02.* [*Archived*](https://web.archive.org/web/20140902081954/https:/dev.opera.com/blog/introducing-opera-15-for-desktop-and-a-fast-release-cycle/) *from the original on 2014-09-02. Retrieved 2014-09-23.*

  same as Chrome 26–29

  same as Chrome 30 and later

  same as Chrome 33 and later

  *Microsoft (2012-09-05).* [*"Secure Channel"*](https://msdn.microsoft.com/en-us/library/aa380123.aspx)*.* [*Archived*](https://web.archive.org/web/20120829025314/http:/msdn.microsoft.com/en-us/library/aa380123.aspx) *from the original on 2012-08-29. Retrieved 2012-10-18.*

  *Microsoft (2009-02-27).* [*"MS-TLSP Appendix A"*](https://msdn.microsoft.com/en-us/library/dd208005.aspx)*.* [*Archived*](https://web.archive.org/web/20130927055954/http:/msdn.microsoft.com/en-us/library/dd208005.aspx) *from the original on 2013-09-27. Retrieved 2009-03-19.*

  [*"What browsers only support SSLv2?"*](https://stackoverflow.com/q/881563)*. Retrieved 2014-06-19.*

  [*"SHA2 and Windows – Windows PKI blog – Site Home – TechNet Blogs"*](https://blogs.technet.com/b/pki/archive/2010/09/30/sha2-and-windows.aspx)*. 2010-09-30.* [*Archived*](https://web.archive.org/web/20140716230537/http:/blogs.technet.com/b/pki/archive/2010/09/30/sha2-and-windows.aspx) *from the original on 2014-07-16. Retrieved 2014-07-29.*

  [*"HTTPS Security Improvements in Internet Explorer 7"*](https://msdn.microsoft.com/en-us/library/bb250503.aspx)*.* [*Archived*](https://web.archive.org/web/20131010205312/http:/msdn.microsoft.com/en-us/library/bb250503.aspx) *from the original on 2013-10-10. Retrieved 2013-10-29.*

  [*"TLS Cipher Suites"*](https://msdn.microsoft.com/en-us/library/windows/desktop/aa380512.aspx)*. Microsoft.* [*Archived*](https://web.archive.org/web/20170313125201/https:/msdn.microsoft.com/en-us/library/windows/desktop/aa380512.aspx) *from the original on 2017-03-13.*

  [*"Archived copy"*](https://support.microsoft.com/kb/948963)*.* [*Archived*](https://web.archive.org/web/20150311105145/http:/support.microsoft.com/kb/948963) *from the original on 2015-03-11. Retrieved 2017-07-19.*

  [*"Vulnerability in Schannel Could Allow Security Feature Bypass (3046049)"*](https://technet.microsoft.com/en-us/security/MS15-031)*. 2015-03-10.* [*Archived*](https://web.archive.org/web/20170313130837/https:/technet.microsoft.com/en-us/security/MS15-031) *from the original on 2017-03-13. Retrieved 2015-03-11.*

  [*"Vulnerability in Schannel Could Allow Information Disclosure (3061518)"*](https://technet.microsoft.com/en-us/security/MS15-055)*. 2015-05-12.* [*Archived*](https://web.archive.org/web/20161008035505/https:/technet.microsoft.com/en-us/security/MS15-055) *from the original on 2016-10-08. Retrieved 2015-05-22.*

  [*"Update to add support for TLS 1.1 and TLS 1.2 in Windows Server 2008 SP2, Windows Embedded POSReady 2009, and Windows Embedded Standard 2009"*](https://support.microsoft.com/kb/4019276)*. Retrieved 2017-07-19.*

  [*"Windows 7 adds support for TLSv1.1 and TLSv1.2 – IEInternals – Site Home – MSDN Blogs"*](https://blogs.msdn.com/b/ieinternals/archive/2009/06/19/windows-7-support-for-tls-and-ciphers.aspx)*.* [*Archived*](https://web.archive.org/web/20131226222853/http:/blogs.msdn.com/b/ieinternals/archive/2009/06/19/windows-7-support-for-tls-and-ciphers.aspx) *from the original on 2013-12-26. Retrieved 2013-10-29.*

  *Thomlinson, Matt (2014-11-11).* [*"Hundreds of Millions of Microsoft Customers Now Benefit from Best-in-Class Encryption"*](https://blogs.microsoft.com/cybertrust/2014/11/11/hundreds-of-millions-of-microsoft-customers-now-benefit-from-best-in-class-encryption/)*. Microsoft Security.* [*Archived*](https://web.archive.org/web/20141114143813/http:/blogs.microsoft.com/cybertrust/2014/11/11/hundreds-of-millions-of-microsoft-customers-now-benefit-from-best-in-class-encryption/) *from the original on 2014-11-14. Retrieved 2014-11-14.*

  [Microsoft security advisory: Update for disabling RC4](https://support.microsoft.com/kb/2868725/) [Archived](https://web.archive.org/web/20150311053342/http:/support.microsoft.com/kb/2868725) 2015-03-11 at the [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine)

  *Microsoft (2013-09-24).* [*"IE11 Changes"*](https://blogs.msdn.com/b/ieinternals/archive/2013/09/24/internet-explorer-11-changelist-change-log.aspx)*.* [*Archived*](https://web.archive.org/web/20131030114356/http:/blogs.msdn.com/b/ieinternals/archive/2013/09/24/internet-explorer-11-changelist-change-log.aspx) *from the original on 2013-10-30. Retrieved 2013-11-01.*

  [*"February 2015 security updates for Internet Explorer"*](https://blogs.msdn.com/b/ie/archive/2015/02/10/february-2015-security-updates-for-internet-explorer.aspx)*. 2015-02-11.* [*Archived*](https://web.archive.org/web/20150211031724/http:/blogs.msdn.com/b/ie/archive/2015/02/10/february-2015-security-updates-for-internet-explorer.aspx) *from the original on 2015-02-11. Retrieved 2015-02-11.*

  [*"Update turns on the setting to disable SSL 3.0 fallback for protected mode sites by default in Internet Explorer 11"*](https://support.microsoft.com/kb/3038778)*.* [*Archived*](https://web.archive.org/web/20150214082207/http:/support.microsoft.com/kb/3038778) *from the original on 2015-02-14. Retrieved 2015-02-11.*

  [*"Vulnerability in SSL 3.0 Could Allow Information Disclosure"*](https://technet.microsoft.com/en-us/security/3009008)*. 2015-04-14.* [*Archived*](https://web.archive.org/web/20161008035547/https:/technet.microsoft.com/en-us/security/3009008) *from the original on 2016-10-08. Retrieved 2015-04-14.*

  *Microsoft Edge Team (2016-08-09).* [*"RC4 is now disabled in Microsoft Edge and Internet Explorer 11"*](https://blogs.windows.com/msedgedev/2016/08/09/rc4-now-deprecated/)*. Microsoft.* [*Archived*](https://web.archive.org/web/20160821091621/https:/blogs.windows.com/msedgedev/2016/08/09/rc4-now-deprecated/) *from the original on 2016-08-21.*

  [*"Internet Explorer 11 for Windows Server 2012 and Windows Embedded 8 Standard"*](https://support.microsoft.com/en-us/help/4492872/update-for-internet-explorer-april-16-2019)*. Microsoft Support. 2019-04-16.*

  [*"TLS (Schannel SSP) changes in Windows 10 and Windows Server 2016"*](https://web.archive.org/web/20170330011044/https:/technet.microsoft.com/en-us/windows-server-docs/security/tls/tls-schannel-ssp-changes-in-windows-10-and-windows-server-2016)*. Microsoft. 2017-03-21. Archived from* [*the original*](https://technet.microsoft.com/en-us/windows-server-docs/security/tls/tls-schannel-ssp-changes-in-windows-10-and-windows-server-2016) *on 2017-03-30. Retrieved 2017-03-29.*

  <https://docs.microsoft.com/en-us/windows/whats-new/whats-new-windows-10-version-1909#security>

  [*"What browsers work with Universal SSL"*](https://support.cloudflare.com/hc/en-us/articles/203041594-What-browsers-work-with-Universal-SSL)*.* [*Archived*](https://web.archive.org/web/20160304023941/https:/support.cloudflare.com/hc/en-us/articles/203041594-What-browsers-work-with-Universal-SSL) *from the original on 2016-03-04. Retrieved 2015-06-15.*

  [*"POODLE SSL vulnerability – secure your Windo… – Windows Phone 8 Development and Hacking"*](http://forum.xda-developers.com/windows-phone-8/development/poodle-ssl-vulnerability-secure-windows-t2906203)*. XDA Developers.* [*Archived*](https://web.archive.org/web/20160923053812/http:/forum.xda-developers.com/windows-phone-8/development/poodle-ssl-vulnerability-secure-windows-t2906203) *from the original on 2016-09-23.*

  [*"What TLS version is used in Windows Phone 8 for secure HTTP connections?"*](https://social.msdn.microsoft.com/Forums/windowsapps/en-US/2ddee177-5086-4453-987b-d02b6a7ec62d/what-tls-version-is-used-in-windows-phone-8-for-secure-http-connections?forum=wpdevelop)*. Microsoft.* [*Archived*](https://web.archive.org/web/20160304063257/https:/social.msdn.microsoft.com/Forums/windowsapps/en-US/2ddee177-5086-4453-987b-d02b6a7ec62d/what-tls-version-is-used-in-windows-phone-8-for-secure-http-connections?forum=wpdevelop) *from the original on 2016-03-04. Retrieved 2014-11-07.*

  [*"Qualys SSL Labs – Projects / User Agent Capabilities: Unknown"*](https://www.ssllabs.com/ssltest/viewClient.html?name=IE%20Mobile&version=10&platform=Win%20Phone%208.0)*.* [*Archived*](https://web.archive.org/web/20170301141459/https:/www.ssllabs.com/ssltest/viewClient.html?name=IE%20Mobile&version=10&platform=Win%20Phone%208.0) *from the original on 2017-03-01.*

  [*"Platform Security"*](https://technet.microsoft.com/en-us/library/dn756283.aspx)*. Microsoft. 2014-06-25.* [*Archived*](https://web.archive.org/web/20170313125118/https:/technet.microsoft.com/en-us/library/dn756283.aspx) *from the original on 2017-03-13. Retrieved 2014-11-07.*

  [*"Release Notes: Important Issues in Windows 8.1 Preview"*](https://technet.microsoft.com/en-us/library/dn303404.aspx)*. Microsoft. 2013-06-24.* [*Archived*](https://web.archive.org/web/20141104071622/http:/technet.microsoft.com/en-us/library/dn303404.aspx) *from the original on 2014-11-04. Retrieved 2014-11-04.*

  [*"W8.1(IE11) vs RC4"*](https://community.qualys.com/thread/12092)*. Qualys Community.* [*Archived*](https://web.archive.org/web/20141104093736/https:/community.qualys.com/thread/12092) *from the original on 2014-11-04. Retrieved 2014-11-04.*

  *Adrian, Dimcev.* [*"Common browsers/libraries/servers and the associated cipher suites implemented"*](http://www.carbonwind.net/TLS_Cipher_Suites_Project/tls_ssl_cipher_suites_annex_a1_main.docx)*. TLS Cipher Suites Project.* [*Archived*](https://www.webcitation.org/6FwL0CvUM?url=http://www.carbonwind.net/TLS_Cipher_Suites_Project/tls_ssl_cipher_suites_annex_a1_main.docx) *from the original on 2013-04-17.*

  [*"Features"*](https://www.apple.com/safari/features.html)*. Safari. Apple. 2009-06-10.* [*Archived*](https://www.webcitation.org/6FwL0oLMO?url=http://www.apple.com/safari/features.html) *from the original on 2013-04-17. Retrieved 2009-06-10.*

  [*"Curl: Patch to add TLS 1.1 and 1.2 support & replace deprecated functions in SecureTransport"*](http://curl.haxx.se/mail/lib-2012-08/0120.html)*. Sweden: haxx.se.* [*Archived*](https://web.archive.org/web/20170301142904/https:/curl.haxx.se/mail/lib-2012-08/0120.html) *from the original on 2017-03-01.*

  [Qualys SSL Report: google.co.uk](https://www.ssllabs.com/ssltest/analyze.html?d=google.co.uk&s=74.125.227.183) [Archived](https://web.archive.org/web/20170320192352/https:/www.ssllabs.com/ssltest/analyze.html?d=google.co.uk&s=74.125.227.183) 2017-03-20 at the [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine) (simulation Safari 5.1.9 TLS 1.0)

  [*"Apple Secures Mac OS X with Mavericks Release"*](http://www.esecurityplanet.com/mac-os-security/apple-secures-mac-os-x-with-mavericks-release.html)*. eSecurity Planet. 2013-10-25.* [*Archived*](https://web.archive.org/web/20140708195022/http:/www.esecurityplanet.com/mac-os-security/apple-secures-mac-os-x-with-mavericks-release.html) *from the original on 2014-07-08. Retrieved 2014-06-23.*

  *Ristic, Ivan (2013-09-10).* [*"Is BEAST Still a Threat?"*](https://community.qualys.com/blogs/securitylabs/2013/09/10/is-beast-still-a-threat)*. Qualys.* [*Archived*](https://web.archive.org/web/20141012121824/https:/community.qualys.com/blogs/securitylabs/2013/09/10/is-beast-still-a-threat) *from the original on 2014-10-12.*

  *Ristić, Ivan (2013-10-31).* [*"Apple enabled BEAST mitigations in OS X 10.9 Mavericks"*](http://blog.ivanristic.com/2013/10/apple-enabled-beast-mitigations-in-mavericks.html)*.* [*Archived*](https://web.archive.org/web/20131107045223/http:/blog.ivanristic.com/2013/10/apple-enabled-beast-mitigations-in-mavericks.html) *from the original on 2013-11-07. Retrieved 2013-11-07.*

  *Ristić, Ivan (2014-02-26).* [*"Apple finally releases patch for BEAST"*](https://community.qualys.com/thread/12496)*. Qualys.* [*Archived*](https://web.archive.org/web/20140714162556/https:/community.qualys.com/thread/12496) *from the original on 2014-07-14. Retrieved 2014-07-01.*

  [*"About Security Update 2014-005"*](http://support.apple.com/kb/HT6531)*. Apple Support knowledge base article. Apple.* [*Archived*](https://web.archive.org/web/20141024181953/https:/support.apple.com/kb/HT6531) *from the original on 2014-10-24.*

  [*"About the security content of iOS 8.1"*](http://support.apple.com/kb/HT6541)*. Apple Support knowledge base article. Apple.* [*Archived*](https://web.archive.org/web/20141023104948/http:/support.apple.com/kb/HT6541) *from the original on 2014-10-23.*

  [*"About Security Update 2015-002"*](https://support.apple.com/en-us/HT204413)*. Apple Support knowlege base article. Apple.* [*Archived*](https://web.archive.org/web/20150316081731/https:/support.apple.com/en-us/HT204413) *from the original on 2015-03-16. Retrieved 2015-03-09.*

  [*"About the security content of OS X Mavericks v10.9"*](http://support.apple.com/kb/HT6011)*.* [*Archived*](https://web.archive.org/web/20140704220956/http:/support.apple.com/kb/HT6011) *from the original on 2014-07-04. Retrieved 2014-06-20.*

  [*"User Agent Capabilities: Safari 8 / OS X 10.10"*](https://www.ssllabs.com/ssltest/viewClient.html?name=Safari&version=8&platform=OS%20X%2010.10)*. Qualys SSL Labs.* [*Archived*](https://web.archive.org/web/20150906044018/https:/www.ssllabs.com/ssltest/viewClient.html?name=Safari&version=8&platform=OS%20X%2010.10) *from the original on 2015-09-06. Retrieved 2015-03-07.*

  [*"About the security content of OS X Yosemite v10.10.4 and Security Update 2015-005"*](https://support.apple.com/en-us/HT204942)*.* [*Archived*](https://web.archive.org/web/20150702202131/https:/support.apple.com/en-us/HT204942) *from the original on 2015-07-02. Retrieved 2015-07-03.*

  *Pauly, Tommy (2019-01-29).* [*"TLS 1.3 in iOS"*](https://mailarchive.ietf.org/arch/msg/tls/5QjzTilqjomSyzENtgfaAqQOhbA)*. tls@ietf.org (Mailing list).*

  [*"Technical Note TN2287 – iOS 5 and TLS 1.2 Interoperability Issues"*](https://developer.apple.com/library/ios/#technotes/tn2287/_index.html)*. Apple. 2011-10-14.* [*Archived*](https://web.archive.org/web/20110907013839/http:/developer.apple.com/library/iOS/#technotes/tn2287/_index.html) *from the original on 2011-09-07. Retrieved 2012-12-10.*

  *Liebowitz, Matt (2011-10-13).* [*"Apple issues huge software security patches"*](https://www.nbcnews.com/id/44896639)*. NBC News. Retrieved 2012-12-10.*

  *MWR Info Security (2012-04-16).* [*"Adventures with iOS UIWebviews"*](http://labs.mwrinfosecurity.com/blog/2012/04/16/adventures-with-ios-uiwebviews/)*.* [*Archived*](https://www.webcitation.org/6FwL2M2Mc?url=http://labs.mwrinfosecurity.com/blog/2012/04/16/adventures-with-ios-uiwebviews/) *from the original on 2013-04-17. Retrieved 2012-12-10.*, section "HTTPS (SSL/TLS)"

  [*"Secure Transport Reference"*](https://developer.apple.com/library/mac/documentation/security/Reference/secureTransportRef/Reference/reference.html#//apple_ref/c/tdef/SSLProtocol)*.* [*Archived*](https://web.archive.org/web/20140604052511/https:/developer.apple.com/library/Mac/documentation/Security/Reference/secureTransportRef/Reference/reference.html#//apple_ref/c/tdef/SSLProtocol) *from the original on 2014-06-04. Retrieved 2014-06-23.* kSSLProtocol2 is deprecated in iOS

  [*"iPhone 3.0: Mobile Safari Gets Enhanced Security Certificate Visualization"*](https://web.archive.org/web/20090403074546/http:/www.theiphoneblog.com/2009/03/31/iphone-30-mobile-safari-enhanced-security-certificate-visualization/)*. The iPhone Blog. 2009-03-31. Archived from* [*the original*](http://www.theiphoneblog.com/2009/03/31/iphone-30-mobile-safari-enhanced-security-certificate-visualization/) *on 2009-04-03.*

  [*"Projects / User Agent Capabilities: Safari 7 / iOS 7.1"*](https://www.ssllabs.com/ssltest/viewClient.html?name=Safari&version=7&platform=iOS%207.1)*. Qualys SSL Labs.* [*Archived*](https://web.archive.org/web/20170313130545/https:/www.ssllabs.com/ssltest/viewClient.html?name=Safari&version=7&platform=iOS%207.1) *from the original on 2017-03-13.*

  *schurtertom (2013-10-11).* [*"SOAP Request fails randomly on one Server but works on another on iOS7"*](https://stackoverflow.com/q/19221568)*.* [*Stack Overflow*](https://en.wikipedia.org/wiki/Stack_Overflow)*. Retrieved 2014-01-05.*

  [*"User Agent Capabilities: Safari 8 / iOS 8.1.2"*](https://www.ssllabs.com/ssltest/viewClient.html?name=Safari&version=8&platform=iOS%208.1.2)*. Qualys SSL Labs.* [*Archived*](https://web.archive.org/web/20160304062526/https:/www.ssllabs.com/ssltest/viewClient.html?name=Safari&version=8&platform=iOS%208.1.2) *from the original on 2016-03-04. Retrieved 2015-03-07.*

  [*"About the security content of iOS 8.2"*](https://support.apple.com/en-us/HT204423)*. Apple Support knowledge base article. Apple.* [*Archived*](https://web.archive.org/web/20150309201042/https:/support.apple.com/en-us/HT204423) *from the original on 2015-03-09. Retrieved 2015-03-09.*

  [*"About the security content of iOS 8.4"*](https://support.apple.com/en-us/HT204941)*.* [*Archived*](https://web.archive.org/web/20150703014410/https:/support.apple.com/en-us/HT204941) *from the original on 2015-07-03. Retrieved 2015-07-03.*

  [*"SSLSocket | Android Developers"*](https://developer.android.com/reference/javax/net/ssl/SSLSocket.html)*.* [*Archived*](https://web.archive.org/web/20150318121117/http:/developer.android.com/reference/javax/net/ssl/SSLSocket.html) *from the original on 2015-03-18. Retrieved 2015-03-11.*

  [*"SSLSocket | Android Developers"*](https://developer.android.com/reference/javax/net/ssl/SSLSocket.html)*.* [*Archived*](https://web.archive.org/web/20160304202641/https:/developer.android.com/reference/javax/net/ssl/SSLSocket.html) *from the original on 2016-03-04. Retrieved 2015-12-17.*

  [*"Android 5.0 Behavior Changes | Android Developers"*](https://developer.android.com/about/versions/android-5.0-changes.html#ssl)*.* [*Archived*](https://web.archive.org/web/20150309000956/http:/developer.android.com/about/versions/android-5.0-changes.html#ssl) *from the original on 2015-03-09. Retrieved 2015-03-11.*

  [*"Android 8.0 Behavior Changes"*](https://developer.android.com/about/versions/oreo/android-8.0-changes.html)*.* [*Archived*](https://web.archive.org/web/20171201042705/https:/developer.android.com/about/versions/oreo/android-8.0-changes.html) *from the original on 2017-12-01.*

  *Oracle.* [*"7093640: Enable client-side TLS 1.2 by default"*](https://bugs.openjdk.java.net/browse/JDK-7093640)*. Retrieved 2018-08-30.*

  *Oracle.* [*"JEP 332: Transport Layer Security (TLS) 1.3"*](https://bugs.openjdk.java.net/browse/JDK-8145252)*. Retrieved 2018-08-30.*

  [*"Version 1.11.13, 2015-01-11 — Botan"*](https://web.archive.org/web/20150109154102/http:/botan.randombit.net/relnotes/1_11_13.html)*. 2015-01-11. Archived from* [*the original*](http://botan.randombit.net/relnotes/1_11_13.html) *on 2015-01-09. Retrieved 2015-01-16.*

  [*"[gnutls-devel] GnuTLS 3.4.0 released"*](http://lists.gnutls.org/pipermail/gnutls-devel/2015-April/007535.html)*. 2015-04-08.* [*Archived*](https://web.archive.org/web/20150416005543/http:/lists.gnutls.org/pipermail/gnutls-devel/2015-April/007535.html) *from the original on 2015-04-16. Retrieved 2015-04-16.*

  [*"[gnutls-devel] gnutls 3.6.4"*](https://lists.gnupg.org/pipermail/gnutls-help/2018-September/004457.html)*. 2018-09-24. Retrieved 2020-05-18.*

  [*"Java™ SE Development Kit 8, Update 31 Release Notes"*](http://www.oracle.com/technetwork/java/javase/8u31-relnotes-2389094.html)*.* [*Archived*](https://web.archive.org/web/20150121105601/http:/www.oracle.com/technetwork/java/javase/8u31-relnotes-2389094.html) *from the original on 2015-01-21. Retrieved 2015-01-22.*

  [*"OpenBSD 5.6 Released"*](https://marc.info/?l=openbsd-announce&m=141486254309079&w=2)*. 2014-11-01. Retrieved 2015-01-20.*

  [*"LibreSSL 2.3.0 Released"*](https://marc.info/?l=openbsd-announce&m=144304330731220)*. 2015-09-23. Retrieved 2015-09-24.*

  [*"MatrixSSL – News"*](https://web.archive.org/web/20150214105056/http:/www.matrixssl.org/news.html)*. Archived from* [*the original*](http://www.matrixssl.org/news.html) *on 2015-02-14. Retrieved 2014-11-09.*

  [*"mbed TLS 2.0.0 released"*](https://tls.mbed.org/tech-updates/releases/mbedtls-2.0.0-released)*. 2015-07-10.* [*Archived*](https://web.archive.org/web/20150925030547/https:/tls.mbed.org/tech-updates/releases/mbedtls-2.0.0-released) *from the original on 2015-09-25. Retrieved 2015-07-14.*

  [*"NSS 3.19 release notes"*](https://developer.mozilla.org/en-US/docs/Mozilla/Projects/NSS/NSS_3.19_release_notes)*. Mozilla Developer Network. Mozilla.* [*Archived*](https://web.archive.org/web/20150605054647/https:/developer.mozilla.org/en-US/docs/Mozilla/Projects/NSS/NSS_3.19_release_notes) *from the original on 2015-06-05. Retrieved 2015-05-06.*

  [*"NSS 3.14 release notes"*](https://developer.mozilla.org/en-US/docs/NSS/NSS_3.14_release_notes)*. Mozilla Developer Network. Mozilla.* [*Archived*](https://web.archive.org/web/20130117130029/https:/developer.mozilla.org/en-US/docs/NSS/NSS_3.14_release_notes) *from the original on 2013-01-17. Retrieved 2012-10-27.*

  [*"NSS 3.15.1 release notes"*](https://developer.mozilla.org/en-US/docs/NSS/NSS_3.15.1_release_notes)*. Mozilla Developer Network. Mozilla.* [*Archived*](https://web.archive.org/web/20130922223238/https:/developer.mozilla.org/en-US/docs/NSS/NSS_3.15.1_release_notes) *from the original on 2013-09-22. Retrieved 2013-08-10.*

  [*"NSS 3.39 release notes"*](https://developer.mozilla.org/en-US/docs/Mozilla/Projects/NSS/NSS_3.39_release_notes#Notable_Changes_in_NSS_3.39)*. 2018-08-31. Retrieved 2018-09-14.*

  [*"OpenSSL 1.1.0 Series Release Notes"*](https://www.openssl.org/news/openssl-1.1.0-notes.html)*.* [*Archived*](https://web.archive.org/web/20160825171308/https:/www.openssl.org/news/openssl-1.1.0-notes.html) *from the original on 2016-08-25. Retrieved 2016-10-02.*

  [*"Major changes between OpenSSL 1.0.0h and OpenSSL 1.0.1 [14 Mar 2012]"*](https://web.archive.org/web/20150120120428/https:/www.openssl.org/news/openssl-1.0.1-notes.html)*. 2012-03-14. Archived from* [*the original*](https://www.openssl.org/news/openssl-1.0.1-notes.html) *on January 20, 2015. Retrieved 2015-01-20.*

  [*"OpenSSL 1.1.1 Is Released"*](https://www.openssl.org/blog/blog/2018/09/11/release111/)*. 2018-09-11. Retrieved 2018-09-14.*

  [TLS cipher suites in Microsoft Windows XP and 2003](https://msdn.microsoft.com/en-us/library/windows/desktop/aa380512%28v=vs.85%29.aspx) [Archived](https://web.archive.org/web/20150118053924/http:/msdn.microsoft.com/en-us/library/windows/desktop/aa380512(v=vs.85).aspx) 2015-01-18 at the [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine)

  [SChannel Cipher Suites in Microsoft Windows Vista](https://msdn.microsoft.com/en-us/library/windows/desktop/ff468651%28v=vs.85%29.aspx) [Archived](https://web.archive.org/web/20150112014148/http:/msdn.microsoft.com/en-us/library/windows/desktop/ff468651(v=vs.85).aspx) 2015-01-12 at the [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine)

  [TLS Cipher Suites in SChannel for Windows 7, 2008R2, 8, 2012](https://msdn.microsoft.com/en-us/library/windows/desktop/aa374757%28v=vs.85%29.aspx) [Archived](https://web.archive.org/web/20150319043951/https:/msdn.microsoft.com/en-us/library/windows/desktop/aa374757(v=vs.85).aspx) 2015-03-19 at the [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine)

  [*"[wolfssl] wolfSSL 3.6.6 Released"*](http://wolfssl.com/wolfSSL/Blog/Entries/2015/8/24_wolfSSL_3.6.6_is_Now_Available.html)*. 2015-08-20.* [*Archived*](https://web.archive.org/web/20151017033726/https:/wolfssl.com/wolfSSL/Blog/Entries/2015/8/24_wolfSSL_3.6.6_is_Now_Available.html) *from the original on 2015-10-17. Retrieved 2015-08-25.*

  [*"[wolfssl] wolfSSL TLS1.3 support"*](https://github.com/wolfSSL/wolfssl/pull/661)*. 2017-02-13. Retrieved 2017-02-13.*

  [*"NSS 3.24 release notes"*](https://developer.mozilla.org/en-US/docs/Mozilla/Projects/NSS/NSS_3.24_release_notes)*. Mozilla Developer Network. Mozilla.* [*Archived*](https://web.archive.org/web/20160826100711/https:/developer.mozilla.org/en-US/docs/Mozilla/Projects/NSS/NSS_3.24_release_notes) *from the original on 2016-08-26. Retrieved 2016-06-19.*

  [*"Technical Note TN2287: iOS 5 and TLS 1.2 Interoperability Issues"*](https://developer.apple.com/library/ios/technotes/tn2287/)*. iOS Developer Library. Apple Inc.* [*Archived*](https://web.archive.org/web/20150403092122/https:/developer.apple.com/library/ios/technotes/tn2287/) *from the original on 2015-04-03. Retrieved 2012-05-03.*

  [Qualys SSL Labs – Projects / User Agent Capabilities](https://dev.ssllabs.com/ssltest/clients.html) [Archived](https://web.archive.org/web/20150919073935/https:/dev.ssllabs.com/ssltest/clients.html) 2015-09-19 at the [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine)

  *Georgiev, Martin and Iyengar, Subodh and Jana, Suman and Anubhai, Rishita and Boneh, Dan and Shmatikov, Vitaly (2012).* [*The most dangerous code in the world: validating SSL certificates in non-browser software. Proceedings of the 2012 ACM conference on Computer and communications security*](http://www.cs.utexas.edu/~shmat/shmat_ccs12.pdf) *(PDF). pp. 38–49.* [*ISBN*](https://en.wikipedia.org/wiki/ISBN_(identifier))[*978-1-4503-1651-4*](https://en.wikipedia.org/wiki/Special:BookSources/978-1-4503-1651-4)*.* [*Archived*](https://web.archive.org/web/20171022194807/http:/www.cs.utexas.edu/~shmat/shmat_ccs12.pdf) *(PDF) from the original on 2017-10-22.*

  *Joris Claessens; Valentin Dem; Danny De Cock; Bart Preneel; Joos Vandewalle (2002). "On the Security of Today's Online Electronic Banking Systems". Computers & Security.* ***21*** *(3): 253–265.* [*doi*](https://en.wikipedia.org/wiki/Doi_(identifier))*:*[*10.1016/S0167-4048(02)00312-7*](https://doi.org/10.1016%2FS0167-4048%2802%2900312-7)*.*

  *Lawrence, Eric (2005-10-22).* [*"IEBlog: Upcoming HTTPS Improvements in Internet Explorer 7 Beta 2"*](http://blogs.msdn.com/ie/archive/2005/10/22/483795.aspx)*.* [*MSDN*](https://en.wikipedia.org/wiki/Microsoft_Developer_Network) *Blogs.* [*Archived*](https://www.webcitation.org/6FwL2ooJz?url=http://blogs.msdn.com/b/ie/archive/2005/10/22/483795.aspx) *from the original on 2013-04-17. Retrieved 2007-11-25.*

  [*"Bugzilla@Mozilla — Bug 236933 – Disable SSL2 and other weak ciphers"*](https://bugzilla.mozilla.org/show_bug.cgi?id=236933)*.* [*Mozilla Corporation*](https://en.wikipedia.org/wiki/Mozilla_Corporation)*. Retrieved 2007-11-25.*

  ["Opera 9.5 for Windows Changelog"](http://www.opera.com/docs/changelogs/windows/950/) [Archived](https://web.archive.org/web/20090626142453/http:/www.opera.com/docs/changelogs/windows/950/) 2009-06-26 at the [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine) at [Opera.com](https://en.wikipedia.org/wiki/Opera.com): "Disabled SSL v2 and weak ciphers."

  ["Opera 10 for Windows changelog"](http://www.opera.com/docs/changelogs/windows/1000/) [Archived](https://web.archive.org/web/20130326080606/http:/www.opera.com/docs/changelogs/windows/1000/) 2013-03-26 at the [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine) at [Opera.com](https://en.wikipedia.org/wiki/Opera.com): "Removed support for SSL v2 and weak ciphers"

  *Pettersen, Yngve (2007-04-30).* [*"10 years of SSL in Opera — Implementer's notes"*](https://web.archive.org/web/20071012205124/http:/my.opera.com/yngve/blog/2007/04/30/10-years-of-ssl-in-opera)*.* [*Opera Software*](https://en.wikipedia.org/wiki/Opera_Software)*. Archived from* [*the original*](http://my.opera.com/yngve/blog/2007/04/30/10-years-of-ssl-in-opera) *on October 12, 2007. Retrieved 2007-11-25.*

  *National Institute of Standards and Technology (December 2010).* [*"Implementation Guidance for FIPS PUB 140-2 and the Cryptographic Module Validation Program"*](https://web.archive.org/web/20101106130105/http:/csrc.nist.gov/groups/STM/cmvp/documents/fips140-2/FIPS1402IG.pdf) *(PDF). Archived from* [*the original*](http://csrc.nist.gov/groups/STM/cmvp/documents/fips140-2/FIPS1402IG.pdf) *(PDF) on November 6, 2010.*

  [*"Summarizing Known Attacks on Transport Layer Security (TLS) and Datagram TLS (DTLS)"*](https://tools.ietf.org/html/rfc7457)*.* [*RFC*](https://en.wikipedia.org/wiki/RFC_(identifier))[*7457*](https://tools.ietf.org/html/rfc7457)*.* [*Archived*](https://web.archive.org/web/20160304201813/https:/tools.ietf.org/html/rfc7457) *from the original on 2016-03-04.*

  [*"CVE – CVE-2009-3555"*](http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2009-3555)*.* [*Archived*](https://web.archive.org/web/20160104234608/http:/cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2009-3555) *from the original on 2016-01-04.*

  *Eric Rescorla (2009-11-05).* [*"Understanding the TLS Renegotiation Attack"*](http://www.educatedguesswork.org/2009/11/understanding_the_tls_renegoti.html)*. Educated Guesswork.* [*Archived*](https://www.webcitation.org/65JSTKRCA?url=http://www.educatedguesswork.org/2009/11/understanding_the_tls_renegoti.html) *from the original on 2012-02-09. Retrieved 2009-11-27.*

  [*"SSL\_CTX\_set\_options SECURE\_RENEGOTIATION"*](https://www.openssl.org/docs/ssl/SSL_CTX_set_options.html#SECURE_RENEGOTIATION)*. OpenSSL Docs. 2010-02-25.* [*Archived*](https://web.archive.org/web/20101126121933/http:/openssl.org/docs/ssl/SSL_CTX_set_options.html#SECURE_RENEGOTIATION) *from the original on 2010-11-26. Retrieved 2010-11-18.*

  [*"GnuTLS 2.10.0 released"*](http://article.gmane.org/gmane.network.gnutls.general/2046)*. GnuTLS release notes. 2010-06-25.* [*Archived*](https://www.webcitation.org/65JSUYEDA?url=http://article.gmane.org/gmane.network.gnutls.general/2046) *from the original on 2012-02-09. Retrieved 2011-07-24.*

  [*"NSS 3.12.6 release notes"*](https://web.archive.org/web/20120306184633/https:/developer.mozilla.org/NSS_3.12.6_release_notes)*. NSS release notes. 2010-03-03. Archived from* [*the original*](https://developer.mozilla.org/NSS_3.12.6_release_notes) *on March 6, 2012. Retrieved 2011-07-24.*

  *A. Langley; N. Modadugu; B. Moeller (2010-06-02).* [*"Transport Layer Security (TLS) False Start"*](http://tools.ietf.org/html/draft-bmoeller-tls-falsestart-00)*. Internet Engineering Task Force. IETF.* [*Archived*](https://web.archive.org/web/20130905215608/http:/tools.ietf.org/html/draft-bmoeller-tls-falsestart-00) *from the original on 2013-09-05. Retrieved 2013-07-31.*

  *Gruener, Wolfgang.* [*"False Start: Google Proposes Faster Web, Chrome Supports It Already"*](https://web.archive.org/web/20101007061707/http:/www.conceivablytech.com/3299/products/false-start-google-proposes-faster-web-chrome-supports-it-already/)*. Archived from* [*the original*](http://www.conceivablytech.com/3299/products/false-start-google-proposes-faster-web-chrome-supports-it-already) *on 2010-10-07. Retrieved 2011-03-09.*

  *Smith, Brian.* [*"Limited rollback attacks in False Start and Snap Start"*](http://www.ietf.org/mail-archive/web/tls/current/msg06933.html)*.* [*Archived*](https://web.archive.org/web/20110504014418/http:/www.ietf.org/mail-archive/web/tls/current/msg06933.html) *from the original on 2011-05-04. Retrieved 2011-03-09.*

  *Dimcev, Adrian.* [*"False Start"*](http://www.carbonwind.net/blog/post/Random-SSLTLS-101-False-Start.aspx)*. Random SSL/TLS 101.* [*Archived*](https://web.archive.org/web/20110504060256/http:/www.carbonwind.net/blog/post/Random-SSLTLS-101-False-Start.aspx) *from the original on 2011-05-04. Retrieved 2011-03-09.*

  *Mavrogiannopoulos, Nikos; Vercautern, Frederik; Velichkov, Vesselin; Preneel, Bart (2012).* [*A cross-protocol attack on the TLS protocol. Proceedings of the 2012 ACM conference on Computer and communications security*](https://www.cosic.esat.kuleuven.be/publications/article-2216.pdf) *(PDF). pp. 62–72.* [*ISBN*](https://en.wikipedia.org/wiki/ISBN_(identifier))[*978-1-4503-1651-4*](https://en.wikipedia.org/wiki/Special:BookSources/978-1-4503-1651-4)*.* [*Archived*](https://web.archive.org/web/20150706104327/https:/www.cosic.esat.kuleuven.be/publications/article-2216.pdf) *(PDF) from the original on 2015-07-06.*

  [*"SMACK: State Machine AttaCKs"*](https://www.smacktls.com)*.* [*Archived*](https://web.archive.org/web/20150312074827/https:/www.smacktls.com/) *from the original on 2015-03-12.*

  *Goodin, Dan (2015-05-20).* [*"HTTPS-crippling attack threatens tens of thousands of Web and mail servers"*](https://arstechnica.com/security/2015/05/https-crippling-attack-threatens-tens-of-thousands-of-web-and-mail-servers/)*. Ars Technica.* [*Archived*](https://web.archive.org/web/20170519130937/https:/arstechnica.com/security/2015/05/https-crippling-attack-threatens-tens-of-thousands-of-web-and-mail-servers/) *from the original on 2017-05-19.*

  *Leyden, John (1 March 2016).* [*"One-third of all HTTPS websites open to DROWN attack"*](https://www.theregister.co.uk/2016/03/01/drown_tls_protocol_flaw/)*. The Register.* [*Archived*](https://web.archive.org/web/20160301215536/http:/www.theregister.co.uk/2016/03/01/drown_tls_protocol_flaw/) *from the original on 1 March 2016. Retrieved 2016-03-02.*

  [*"More than 11 million HTTPS websites imperiled by new decryption attack"*](https://arstechnica.com/security/2016/03/more-than-13-million-https-websites-imperiled-by-new-decryption-attack/)*. Ars Technica.* [*Archived*](https://web.archive.org/web/20160301191108/http:/arstechnica.com/security/2016/03/more-than-13-million-https-websites-imperiled-by-new-decryption-attack/) *from the original on 2016-03-01. Retrieved 2016-03-02.*

  *Thai Duong & Juliano Rizzo (2011-05-13).* [*"Here Come The ⊕ Ninjas"*](https://bug665814.bugzilla.mozilla.org/attachment.cgi?id=540839)*.* [*Archived*](https://web.archive.org/web/20140603102506/https:/bug665814.bugzilla.mozilla.org/attachment.cgi?id=540839) *from the original on 2014-06-03.*

  *Dan Goodin (2011-09-19).* [*"Hackers break SSL encryption used by millions of sites"*](https://www.theregister.co.uk/2011/09/19/beast_exploits_paypal_ssl/)*.* [*Archived*](https://www.webcitation.org/65JSSbOus?url=http://www.theregister.co.uk/2011/09/19/beast_exploits_paypal_ssl/) *from the original on 2012-02-09.*

  [*"Y Combinator comments on the issue"*](http://news.ycombinator.com/item?id=3015498)*. 2011-09-20.* [*Archived*](https://www.webcitation.org/6FwL5WZ8b?url=https://news.ycombinator.com/item?id=3015498) *from the original on 2013-04-17.*

  [*"Security of CBC Ciphersuites in SSL/TLS: Problems and Countermeasures"*](https://web.archive.org/web/20120630143111/http:/www.openssl.org/~bodo/tls-cbc.txt)*. 2004-05-20. Archived from* [*the original*](https://www.openssl.org/~bodo/tls-cbc.txt) *on 2012-06-30.*

  *Ristic, Ivan (Sep 10, 2013).* [*"Is BEAST Still a Threat?"*](https://community.qualys.com/blogs/securitylabs/2013/09/10/is-beast-still-a-threat)*.* [*Archived*](https://web.archive.org/web/20141012121824/https:/community.qualys.com/blogs/securitylabs/2013/09/10/is-beast-still-a-threat) *from the original on 12 October 2014. Retrieved 8 October 2014.*

  *Brian Smith (2011-09-30).* [*"(CVE-2011-3389) Rizzo/Duong chosen plaintext attack (BEAST) on SSL/TLS 1.0 (facilitated by websockets -76)"*](https://bugzilla.mozilla.org/show_bug.cgi?id=665814)*.*

  [*"Vulnerability in SSL/TLS Could Allow Information Disclosure (2643584)"*](https://technet.microsoft.com/library/security/ms12-006)*. 2012-01-10.* [*Archived*](https://web.archive.org/web/20140815234322/https:/technet.microsoft.com/library/security/ms12-006) *from the original on 2014-08-15.*

  *Ristic, Ivan (Oct 31, 2013).* [*"Apple Enabled BEAST Mitigations in OS X 10.9 Mavericks"*](https://community.qualys.com/blogs/securitylabs/2013/10/31/apple-enabled-beast-mitigations-in-os-x-109-mavericks)*.* [*Archived*](https://web.archive.org/web/20141012122536/https:/community.qualys.com/blogs/securitylabs/2013/10/31/apple-enabled-beast-mitigations-in-os-x-109-mavericks) *from the original on 12 October 2014. Retrieved 8 October 2014.*

  *Dan Goodin (2012-09-13).* [*"Crack in Internet's foundation of trust allows HTTPS session hijacking"*](https://arstechnica.com/security/2012/09/crime-hijacks-https-sessions/)*. Ars Technica.* [*Archived*](https://web.archive.org/web/20130801104610/http:/arstechnica.com/security/2012/09/crime-hijacks-https-sessions/) *from the original on 2013-08-01. Retrieved 2013-07-31.*

  *Dennis Fisher (September 13, 2012).* [*"CRIME Attack Uses Compression Ratio of TLS Requests as Side Channel to Hijack Secure Sessions"*](https://web.archive.org/web/20120915224635/http:/threatpost.com/en_us/blogs/crime-attack-uses-compression-ratio-tls-requests-side-channel-hijack-secure-sessions-091312)*. ThreatPost. Archived from* [*the original*](http://threatpost.com/en_us/blogs/crime-attack-uses-compression-ratio-tls-requests-side-channel-hijack-secure-sessions-091312) *on September 15, 2012. Retrieved 2012-09-13.*

  *Goodin, Dan (1 August 2013).* [*"Gone in 30 seconds: New attack plucks secrets from HTTPS-protected pages"*](https://arstechnica.com/security/2013/08/gone-in-30-seconds-new-attack-plucks-secrets-from-https-protected-pages/)*. Ars Technica. Condé Nast.* [*Archived*](https://web.archive.org/web/20130803181144/http:/arstechnica.com/security/2013/08/gone-in-30-seconds-new-attack-plucks-secrets-from-https-protected-pages/) *from the original on 3 August 2013. Retrieved 2 August 2013.*

  *Leyden, John (2 August 2013).* [*"Step into the BREACH: New attack developed to read encrypted web data"*](https://www.theregister.co.uk/2013/08/02/breach_crypto_attack/)*. The Register.* [*Archived*](https://web.archive.org/web/20130805233414/http:/www.theregister.co.uk/2013/08/02/breach_crypto_attack/) *from the original on 5 August 2013. Retrieved 2 August 2013.*

  *P. Gutmann (September 2014).* [*"Encrypt-then-MAC for Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS)"*](http://tools.ietf.org/html/rfc7366)*.* [*Archived*](https://web.archive.org/web/20150512214100/https:/tools.ietf.org/html/rfc7366) *from the original on 2015-05-12.*

  *Hagai Bar-El.* [*"Poodle flaw and IoT"*](http://www.hbarel.com/analysis/eng/poodle-flaw-and-iot)*.* [*Archived*](https://web.archive.org/web/20150316030953/https:/www.hbarel.com/analysis/eng/poodle-flaw-and-iot) *from the original on 16 March 2015. Retrieved 15 October 2014.*

  *Langley, Adam (December 8, 2014).* [*"The POODLE bites again"*](https://www.imperialviolet.org/2014/12/08/poodleagain.html)*.* [*Archived*](https://web.archive.org/web/20141208200653/https:/www.imperialviolet.org/2014/12/08/poodleagain.html) *from the original on December 8, 2014. Retrieved 2014-12-08.*

  [security – Safest ciphers to use with the BEAST? (TLS 1.0 exploit) I've read that RC4 is immune – Server Fault](http://serverfault.com/questions/315042/)

  *Pouyan Sepehrdad; Serge Vaudenay; Martin Vuagnoux (2011). "Discovery and Exploitation of New Biases in RC4". In Alex Biryukov; Guang Gong; Douglas R. Stinson (eds.). Selected Areas in Cryptography: 17th International Workshop, SAC 2010, Waterloo, Ontario, Canada, August 12-13, 2010, Revised Selected Papers. Lecture Notes in Computer Science.* ***6544****. pp. 74–91.* [*doi*](https://en.wikipedia.org/wiki/Doi_(identifier))*:*[*10.1007/978-3-642-19574-7\_5*](https://doi.org/10.1007%2F978-3-642-19574-7_5)*.* [*ISBN*](https://en.wikipedia.org/wiki/ISBN_(identifier))[*978-3-642-19573-0*](https://en.wikipedia.org/wiki/Special:BookSources/978-3-642-19573-0)*.*

  *Green, Matthew.* [*"Attack of the week: RC4 is kind of broken in TLS"*](http://blog.cryptographyengineering.com/2013/03/attack-of-week-rc4-is-kind-of-broken-in.html)*. Cryptography Engineering.* [*Archived*](https://web.archive.org/web/20130314214026/http:/blog.cryptographyengineering.com/2013/03/attack-of-week-rc4-is-kind-of-broken-in.html) *from the original on March 14, 2013. Retrieved March 12, 2013.*

  *Nadhem AlFardan, Dan Bernstein, Kenny Paterson, Bertram Poettering and Jacob Schuldt.* [*"On the Security of RC4 in TLS"*](http://www.isg.rhul.ac.uk/tls/)*. Royal Holloway University of London.* [*Archived*](https://web.archive.org/web/20130315084623/http:/www.isg.rhul.ac.uk/tls/) *from the original on March 15, 2013. Retrieved March 13, 2013.*

  *AlFardan, Nadhem J.; Bernstein, Daniel J.; Paterson, Kenneth G.; Poettering, Bertram; Schuldt, Jacob C. N. (8 July 2013).* [*"On the Security of RC4 in TLS and WPA"*](http://www.isg.rhul.ac.uk/tls/RC4biases.pdf) *(PDF).* [*Archived*](https://web.archive.org/web/20130922170155/http:/www.isg.rhul.ac.uk/tls/RC4biases.pdf) *(PDF) from the original on 22 September 2013. Retrieved 2 September 2013.*

  *AlFardan, Nadhem J.; Bernstein, Daniel J.; Paterson, Kenneth G.; Poettering, Bertram; Schuldt, Jacob C. N. (15 August 2013).* [*On the Security of RC4 in TLS*](https://www.usenix.org/sites/default/files/conference/protected-files/alfardan_sec13_slides.pdf) *(PDF). 22nd* [*USENIX*](https://en.wikipedia.org/wiki/USENIX) *Security Symposium. p. 51.* [*Archived*](https://web.archive.org/web/20130922133950/https:/www.usenix.org/sites/default/files/conference/protected-files/alfardan_sec13_slides.pdf) *(PDF) from the original on 22 September 2013. Retrieved 2 September 2013. Plaintext recovery attacks against RC4 in TLS are feasible although not truly practical*

  *Goodin, Dan.* [*"Once-theoretical crypto attack against HTTPS now verges on practicality"*](https://arstechnica.com/security/2015/07/once-theoretical-crypto-attack-against-https-now-verges-on-practicality/)*. Ars Technical. Conde Nast.* [*Archived*](https://web.archive.org/web/20150716084138/http:/arstechnica.com/security/2015/07/once-theoretical-crypto-attack-against-https-now-verges-on-practicality/) *from the original on 16 July 2015. Retrieved 16 July 2015.*

  [*"Mozilla Security Server Side TLS Recommended Configurations"*](https://wiki.mozilla.org/Security/Server_Side_TLS)*. Mozilla.* [*Archived*](https://web.archive.org/web/20150103093047/https:/wiki.mozilla.org/Security/Server_Side_TLS) *from the original on 2015-01-03. Retrieved 2015-01-03.*

  [*"Security Advisory 2868725: Recommendation to disable RC4"*](http://blogs.technet.com/b/srd/archive/2013/11/12/security-advisory-2868725-recommendation-to-disable-rc4.aspx)*. Microsoft. 2013-11-12.* [*Archived*](https://web.archive.org/web/20131118081816/http:/blogs.technet.com/b/srd/archive/2013/11/12/security-advisory-2868725-recommendation-to-disable-rc4.aspx) *from the original on 2013-11-18. Retrieved 2013-12-04.*

  [*"Ending support for the RC4 cipher in Microsoft Edge and Internet Explorer 11"*](https://blogs.windows.com/msedgedev/2015/09/01/ending-support-for-the-rc4-cipher-in-microsoft-edge-and-internet-explorer-11/)*. Microsoft Edge Team. September 1, 2015.* [*Archived*](https://web.archive.org/web/20150902054341/http:/blogs.windows.com/msedgedev/2015/09/01/ending-support-for-the-rc4-cipher-in-microsoft-edge-and-internet-explorer-11/) *from the original on September 2, 2015.*

  *Langley, Adam (Sep 1, 2015).* [*"Intent to deprecate: RC4"*](https://groups.google.com/a/chromium.org/forum/#!msg/security-dev/kVfCywocUO8/vgi_rQuhKgAJ)*.*

  *Barnes, Richard (Sep 1, 2015).* [*"Intent to ship: RC4 disabled by default in Firefox 44"*](https://groups.google.com/forum/#!topic/mozilla.dev.platform/JIEFcrGhqSM/discussion)*.* [*Archived*](http://arquivo.pt/wayback/20110122130054/https:/groups.google.com/forum/#!topic/mozilla.dev.platform/JIEFcrGhqSM/discussion) *from the original on 2011-01-22.*

  *John Leyden (1 August 2013).* [*"Gmail, Outlook.com and e-voting 'pwned' on stage in crypto-dodge hack"*](https://www.theregister.co.uk/2013/08/01/gmail_hotmail_hijacking/)*. The Register.* [*Archived*](https://web.archive.org/web/20130801193054/http:/www.theregister.co.uk/2013/08/01/gmail_hotmail_hijacking/) *from the original on 1 August 2013. Retrieved 1 August 2013.*

  [*"BlackHat USA Briefings"*](https://www.blackhat.com/us-13/briefings.html#Smyth)*. Black Hat 2013.* [*Archived*](https://web.archive.org/web/20130730124037/http:/www.blackhat.com/us-13/briefings.html#Smyth) *from the original on 30 July 2013. Retrieved 1 August 2013.*

  *Smyth, Ben; Pironti, Alfredo (2013).* [*"Truncating TLS Connections to Violate Beliefs in Web Applications"*](https://hal.inria.fr/hal-01102013)*. 7th USENIX Workshop on Offensive Technologies.* [*Archived*](https://web.archive.org/web/20151106110117/https:/hal.inria.fr/hal-01102013) *from the original on 6 November 2015. Retrieved 15 February 2016.*

  *Goodin, Dan.* [*"New attack bypasses HTTPS protection on Macs, Windows, and Linux"*](https://arstechnica.com/security/2016/07/new-attack-that-cripples-https-crypto-works-on-macs-windows-and-linux/)*. Ars Technica. Condé Nast.* [*Archived*](https://web.archive.org/web/20160727160434/http:/arstechnica.com/security/2016/07/new-attack-that-cripples-https-crypto-works-on-macs-windows-and-linux/) *from the original on 27 July 2016. Retrieved 28 July 2016.*

  *Goodin, Dan (August 24, 2016).* [*"HTTPS and OpenVPN face new attack that can decrypt secret cookies"*](https://arstechnica.com/security/2016/08/new-attack-can-pluck-secrets-from-1-of-https-traffic-affects-top-sites/)*. Ars Technica.* [*Archived*](https://web.archive.org/web/20160824181630/http:/arstechnica.com/security/2016/08/new-attack-can-pluck-secrets-from-1-of-https-traffic-affects-top-sites/) *from the original on August 24, 2016. Retrieved August 24, 2016.*

  [*"Why is it called the 'Heartbleed Bug'?"*](https://www.washingtonpost.com/blogs/style-blog/wp/2014/04/09/why-is-it-called-the-heartbleed-bug/)*. The Washington Post. 2014-04-09.* [*Archived*](https://web.archive.org/web/20141009063758/http:/www.washingtonpost.com/blogs/style-blog/wp/2014/04/09/why-is-it-called-the-heartbleed-bug/) *from the original on 2014-10-09.*

  [*"Heartbleed Bug vulnerability [9 April 2014]"*](https://blogs.comodo.com/e-commerce/heartbleed-bug-comodo-urges-openssl-users-to-apply-patch/)*.* [*Comodo Group*](https://en.wikipedia.org/wiki/Comodo_Group)*.* [*Archived*](https://web.archive.org/web/20140705212748/https:/blogs.comodo.com/e-commerce/heartbleed-bug-comodo-urges-openssl-users-to-apply-patch/) *from the original on 5 July 2014.*

  *Bleichenbacher, Daniel (August 2006).* [*"Bleichenbacher's RSA signature forgery based on implementation error"*](https://web.archive.org/web/20141216203704/http:/www.imc.org/ietf-openpgp/mail-archive/msg06063.html)*. Archived from* [*the original*](http://www.imc.org/ietf-openpgp/mail-archive/msg06063.html) *on 2014-12-16.*

  [*"BERserk"*](http://www.intelsecurity.com/advanced-threat-research/)*. Intel Security: Advanced Threat Research. September 2014.* [*Archived*](https://web.archive.org/web/20150112153121/http:/www.intelsecurity.com/advanced-threat-research/) *from the original on 2015-01-12.*

  *Goodin, Dan (February 19, 2015).* [*"Lenovo PCs ship with man-in-the-middle adware that breaks HTTPS connections"*](https://arstechnica.com/information-technology/2015/02/lenovo-pcs-ship-with-man-in-the-middle-adware-that-breaks-https-connections/)*.* [*Ars Technica*](https://en.wikipedia.org/wiki/Ars_Technica)*.* [*Archived*](https://web.archive.org/web/20170912103610/https:/arstechnica.com/information-technology/2015/02/lenovo-pcs-ship-with-man-in-the-middle-adware-that-breaks-https-connections/) *from the original on September 12, 2017. Retrieved December 10, 2017.*

  *Valsorda, Filippo (2015-02-20).* [*"Komodia/Superfish SSL validation is broken"*](https://blog.filippo.io/komodia-superfish-ssl-validation-is-broken/)*. Filippo.io.* [*Archived*](https://web.archive.org/web/20150224112141/https:/blog.filippo.io/komodia-superfish-ssl-validation-is-broken/) *from the original on 2015-02-24.*

  *Goodin, Dan.* [*""Forbidden attack" makes dozens of HTTPS Visa sites vulnerable to tampering"*](https://arstechnica.com/security/2016/05/faulty-https-settings-leave-dozens-of-visa-sites-vulnerable-to-forgery-attacks/)*. Ars Technica.* [*Archived*](https://web.archive.org/web/20160526175713/http:/arstechnica.com/security/2016/05/faulty-https-settings-leave-dozens-of-visa-sites-vulnerable-to-forgery-attacks/) *from the original on 26 May 2016. Retrieved 26 May 2016.*

  *Clark Estes, Adam.* [*"Everything You Need to Know About Cloudbleed, the Latest Internet Security Disaster"*](https://gizmodo.com/everything-you-need-to-know-about-cloudbleed-the-lates-1792710616)*.* [*Gizmodo*](https://en.wikipedia.org/wiki/Gizmodo)*.* [*Archived*](https://web.archive.org/web/20170225013516/http:/gizmodo.com/everything-you-need-to-know-about-cloudbleed-the-lates-1792710616) *from the original on 2017-02-25. Retrieved 2017-02-24.*

  *Diffie, Whitfield; van Oorschot, Paul C; Wiener, Michael J. (June 1992).* [*"Authentication and Authenticated Key Exchanges"*](http://citeseer.ist.psu.edu/diffie92authentication.html)*. Designs, Codes and Cryptography.* ***2*** *(2): 107–125.* [*CiteSeerX*](https://en.wikipedia.org/wiki/CiteSeerX_(identifier))[*10.1.1.59.6682*](https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.59.6682)*.* [*doi*](https://en.wikipedia.org/wiki/Doi_(identifier))*:*[*10.1007/BF00124891*](https://doi.org/10.1007%2FBF00124891)*.* [*Archived*](https://web.archive.org/web/20080313081157/http:/citeseer.ist.psu.edu/diffie92authentication.html) *from the original on 2008-03-13. Retrieved 2008-02-11.*

  [Discussion on the TLS mailing list in October 2007](http://www1.ietf.org/mail-archive/web/tls/current/msg02134.html) [Archived](https://web.archive.org/web/20130922103746/http:/www.ietf.org/mail-archive/web/tls/current/msg02134.html) 2013-09-22 at the [Wayback Machine](https://en.wikipedia.org/wiki/Wayback_Machine)

  [*"Protecting data for the long term with forward secrecy"*](http://googleonlinesecurity.blogspot.com.au/2011/11/protecting-data-for-long-term-with.html)*.* [*Archived*](https://web.archive.org/web/20130506184654/http:/googleonlinesecurity.blogspot.com.au/2011/11/protecting-data-for-long-term-with.html) *from the original on 2013-05-06. Retrieved 2012-11-05.*

  *Bernat, Vincent.* [*"SSL/TLS & Perfect Forward Secrecy"*](https://vincent.bernat.im/en/blog/2011-ssl-perfect-forward-secrecy.html)*.* [*Archived*](https://web.archive.org/web/20120827064047/http:/vincent.bernat.im/en/blog/2011-ssl-perfect-forward-secrecy.html) *from the original on 2012-08-27. Retrieved 2012-11-05.*

  [*"SSL Labs: Deploying Forward Secrecy"*](https://community.qualys.com/blogs/securitylabs/2013/06/25/ssl-labs-deploying-forward-secrecy)*. Qualys.com. 2013-06-25.* [*Archived*](https://web.archive.org/web/20130626193314/https:/community.qualys.com/blogs/securitylabs/2013/06/25/ssl-labs-deploying-forward-secrecy) *from the original on 2013-06-26. Retrieved 2013-07-10.*

  *Ristic, Ivan (2013-08-05).* [*"SSL Labs: Deploying Forward Secrecy"*](https://community.qualys.com/blogs/securitylabs/2013/06/25/ssl-labs-deploying-forward-secrecy)*. Qualsys.* [*Archived*](https://web.archive.org/web/20130920150259/https:/community.qualys.com/blogs/securitylabs/2013/06/25/ssl-labs-deploying-forward-secrecy) *from the original on 2013-09-20. Retrieved 2013-08-31.*

  *Langley, Adam (27 June 2013).* [*"How to botch TLS forward secrecy"*](https://www.imperialviolet.org/2013/06/27/botchingpfs.html)*. imperialviolet.org.* [*Archived*](https://web.archive.org/web/20130808221614/https:/www.imperialviolet.org/2013/06/27/botchingpfs.html) *from the original on 8 August 2013.*

  *Daignière, Florent.* [*"TLS "Secrets": Whitepaper presenting the security implications of the deployment of session tickets (RFC 5077) as implemented in OpenSSL"*](https://media.blackhat.com/us-13/US-13-Daigniere-TLS-Secrets-WP.pdf) *(PDF). Matta Consulting Limited.* [*Archived*](https://web.archive.org/web/20130806233112/https:/media.blackhat.com/us-13/US-13-Daigniere-TLS-Secrets-WP.pdf) *(PDF) from the original on 6 August 2013. Retrieved 7 August 2013.*

  *Daignière, Florent.* [*"TLS "Secrets": What everyone forgot to tell you..."*](https://media.blackhat.com/us-13/US-13-Daigniere-TLS-Secrets-Slides.pdf) *(PDF). Matta Consulting Limited.* [*Archived*](https://web.archive.org/web/20130805134805/https:/media.blackhat.com/us-13/US-13-Daigniere-TLS-Secrets-Slides.pdf) *(PDF) from the original on 5 August 2013. Retrieved 7 August 2013.*

  *L.S. Huang; S. Adhikarla; D. Boneh; C. Jackson (2014).* [*"An Experimental Study of TLS Forward Secrecy Deployments"*](http://crypto.stanford.edu/~dabo/pubs/abstracts/websec_ecc.html)*. IEEE Internet Computing.* ***18*** *(6): 43–51.* [*CiteSeerX*](https://en.wikipedia.org/wiki/CiteSeerX_(identifier))[*10.1.1.663.4653*](https://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.663.4653)*.* [*doi*](https://en.wikipedia.org/wiki/Doi_(identifier))*:*[*10.1109/MIC.2014.86*](https://doi.org/10.1109%2FMIC.2014.86)*.* [*Archived*](https://web.archive.org/web/20150920011317/http:/crypto.stanford.edu/~dabo/pubs/abstracts/websec_ecc.html) *from the original on 20 September 2015. Retrieved 16 October 2015.*

  [*"Protecting data for the long term with forward secrecy"*](http://googleonlinesecurity.blogspot.com.au/2011/11/protecting-data-for-long-term-with.html)*.* [*Archived*](https://web.archive.org/web/20140212214518/http:/googleonlinesecurity.blogspot.com.au/2011/11/protecting-data-for-long-term-with.html) *from the original on 2014-02-12. Retrieved 2014-03-07.*

  *Hoffman-Andrews, Jacob.* [*"Forward Secrecy at Twitter"*](https://blog.twitter.com/2013/forward-secrecy-at-twitter-0)*. Twitter.* [*Archived*](https://web.archive.org/web/20140216041202/https:/blog.twitter.com/2013/forward-secrecy-at-twitter-0) *from the original on 2014-02-16. Retrieved 2014-03-07.*

  *Durumeric, Zakir; Ma, Zane; Springall, Drew; Barnes, Richard; Sullivan, Nick; Bursztein, Elie; Bailey, Michael; Halderman, J. Alex; Paxson, Vern (5 September 2017).* [*"The Security Impact of HTTPS Interception"*](https://www.ndss-symposium.org/ndss2017/ndss-2017-programme/security-impact-https-interception/)*. NDSS Symposium.* [*doi*](https://en.wikipedia.org/wiki/Doi_(identifier))*:*[*10.14722/ndss.2017.23456*](https://doi.org/10.14722%2Fndss.2017.23456)*.* [*ISBN*](https://en.wikipedia.org/wiki/ISBN_(identifier))[*978-1-891562-46-4*](https://en.wikipedia.org/wiki/Special:BookSources/978-1-891562-46-4)*.*

  These certificates are currently [X.509](https://en.wikipedia.org/wiki/X.509), but [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [6091](https://tools.ietf.org/html/rfc6091) also specifies the use of [OpenPGP](https://en.wikipedia.org/wiki/OpenPGP)-based certificates.

  *Chris (2009-02-18).* [*"vsftpd-2.1.0 released – Using TLS session resume for FTPS data connection authentication"*](http://scarybeastsecurity.blogspot.com/2009/02/vsftpd-210-released.html)*. Scarybeastsecurity. blogspot.com.* [*Archived*](https://web.archive.org/web/20120707213409/http:/scarybeastsecurity.blogspot.com/2009/02/vsftpd-210-released.html) *from the original on 2012-07-07. Retrieved 2012-05-17.*

  [*Wildcard SSL Certificate overview*](https://ssl.comodo.com/wildcard-ssl-certificates.php)*,* [*archived*](https://web.archive.org/web/20150623231035/https:/ssl.comodo.com/wildcard-ssl-certificates.php) *from the original on 2015-06-23, retrieved 2015-07-02*

* 1.  [*Named-based SSL virtual hosts: how to tackle the problem*](https://www.switch.ch/pki/meetings/2007-01/namebased_ssl_virtualhosts.pdf) *(PDF),* [*archived*](https://web.archive.org/web/20120803022659/https:/www.switch.ch/pki/meetings/2007-01/namebased_ssl_virtualhosts.pdf) *(PDF) from the original on 2012-08-03, retrieved 2012-05-17*

This article is based on material taken from the [*Free On-line Dictionary of Computing*](https://en.wikipedia.org/wiki/Free_On-line_Dictionary_of_Computing) prior to 1 November 2008 and incorporated under the "relicensing" terms of the [GFDL](https://en.wikipedia.org/wiki/GNU_Free_Documentation_License), version 1.3 or later.

**Further reading**

* *Wagner, David; Schneier, Bruce (November 1996).* [*"Analysis of the SSL 3.0 Protocol"*](http://www.schneier.com/paper-ssl.pdf) *(PDF). The Second USENIX Workshop on Electronic Commerce Proceedings. USENIX Press. pp. 29–40.*
* *Eric Rescorla (2001).* [*SSL and TLS: Designing and Building Secure Systems*](https://archive.org/details/ssltls00eric)*. United States: Addison-Wesley Pub Co.* [*ISBN*](https://en.wikipedia.org/wiki/ISBN_(identifier))[*978-0-201-61598-2*](https://en.wikipedia.org/wiki/Special:BookSources/978-0-201-61598-2)*.*
* *Stephen A. Thomas (2000). SSL and TLS essentials securing the Web. New York: Wiley.* [*ISBN*](https://en.wikipedia.org/wiki/ISBN_(identifier))[*978-0-471-38354-3*](https://en.wikipedia.org/wiki/Special:BookSources/978-0-471-38354-3)*.*
* *Bard, Gregory (2006).* [*"A Challenging But Feasible Blockwise-Adaptive Chosen-Plaintext Attack on SSL"*](http://eprint.iacr.org/2006/136)*. International Association for Cryptologic Research (136). Retrieved 2011-09-23.*
* *Canvel, Brice.* [*"Password Interception in a SSL/TLS Channel"*](http://lasecwww.epfl.ch/memo/memo_ssl.shtml)*. Retrieved 2007-04-20.*
* *IETF Multiple Authors.* [*"RFC of change for TLS Renegotiation"*](http://tools.ietf.org/html/rfc5746)*. Retrieved 2009-12-11.*
* [Creating VPNs with IPsec and SSL/TLS](http://www.linuxjournal.com/article/9916) Linux Journal article by Rami Rosen
* *Polk, Tim; McKay, Kerry; Chokhani, Santosh (April 2014).* [*"Guidelines for the Selection, Configuration, and Use of Transport Layer Security (TLS) Implementations"*](https://web.archive.org/web/20140508025330/http:/nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-52r1.pdf) *(PDF). National Institute of Standards and Technology. Archived from* [*the original*](http://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-52r1.pdf) *(PDF) on 2014-05-08. Retrieved 2014-05-07.*
* *Abdou, AbdelRahman; van Oorschot, Paul (August 2017).* [*"Server Location Verification (SLV) and Server Location Pinning: Augmenting TLS Authentication"*](https://dl.acm.org/citation.cfm?id=3139294)*. Transactions on Privacy and Security. ACM.*

**External links**

|  |  |
| --- | --- |
|  | Wikimedia Commons has media related to [***SSL and TLS***](https://commons.wikimedia.org/wiki/Category:SSL_and_TLS). |

**Specifications** (see [§ Standards](https://en.wikipedia.org/wiki/Transport_Layer_Security#Standards) section for older SSL 2.0, SSL 3.0, TLS 1.0, TLS 1.1 links)

* [The Transport Layer Security (TLS) Protocol Version 1.2](https://tools.ietf.org/html/rfc5246) [RFC](https://en.wikipedia.org/wiki/RFC_(identifier)) [5246](https://tools.ietf.org/html/rfc5246)
* [IETF (Internet Engineering Task Force) TLS Workgroup](https://datatracker.ietf.org/wg/tls/)

TLS version intolerance

* [TLS version intolerance](https://timtaubert.de/blog/2016/09/tls-version-intolerance/)
* [TLS 1.3 and Version Intolerance](https://www.int21.de/slides/berlinsec-versionintolerance/#/)

Other

* [OWASP: Transport Layer Protection Cheat Sheet](https://cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Protection_Cheat_Sheet.html)
* [A talk on SSL/TLS that tries to explain things in terms that people might understand.](https://computing.ece.vt.edu/~jkh/Understanding_SSL_TLS.pdf)
* [SSL: Foundation for Web Security](https://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_1-1/ssl.html)
* [TLS Renegotiation Vulnerability – IETF Tools](https://tools.ietf.org/agenda/76/slides/tls-7.pdf)
* [Trustworthy Internet Movement – SSL Pulse](https://www.trustworthyinternet.org/ssl-pulse/) – Survey of TLS/SSL implementation of the most popular websites
* [How to Generate CSR for SSL](http://www.sslcentral.com/how-to-generate-csr/)
* [How TLS Handshake works in private browser](https://www.gnudeveloper.com/cyber_security/TLS_handshake.html)