Security TLS 1.2

What is TLS?

Transport Layer Security (TLS) and its now deprecated Secure Sockets Layer (SSL) is an encryption protocol designed to keep data safe while transmitting data over the network. The TLS protocol is widely used in applications such as Web browsing, e-mail, instant messaging, and voice over IP, and Internet services such as DNS and NTP. TSL protocol is commonly used among websites applications to ensure the communication security between their servers and web browsers.

It should be noted that TLS does not protect the data on the end system. It just ensures that data is transmitted securely over the Internet, avoiding possible eavesdropping and/or content changes.

The TLS protocol includes two layers: TLS record and TLS handshake protocol. It is normally implemented on top of TCP in order to encrypt Application Layer protocols such as HTTP, FTP, SMTP and IMAP.

History of TLS 1.2

TLS protocol was developed through a joint initiative begun in August 1986. TLS 1.2 was developed based on the earlier TLS version 1.1 specification and was defined in RFC 5246. Comparing to TLS 1.1 and its earlier versions, TLS 1.2 has the major difference includes:

1. It adds an option to use cipher suite specified pseudorandom functions (PRF).
2. It adds an option to use cipher suite hash algorithms with the size of the hash in the finish message still being at least 96 bits.
3. Enhancement in the client’s and server’s ability to specify which hashes and signature algorithms they accept.
4. The MD5-SHA-1 combination in the digitally signed element was replaced with a single hash negotiated during handshake.
5. Expansion of support for authenticated encryption ciphers, used mainly for Galois/Counter Mode (GCM) and CCM mode of Advanced Encryption Standard (AES) encryption.
6. TLS extensions and AES cipher suites were added.
7. Tightened up various requirements.

The greater enhancement of TLS 1.2 encryption makes it possible to use more secure hash algorithms (such as SHA-256) and advanced cipher suites that support elliptic curve encryption. To check if a specific https: // page is encrypted with TLS 1.2, you can run it through ssllabs test. The result will provide you with information about the site's use of security protocols, cipher suites, etc.

How does TLS work?

Since the communication among applications can be done either with or without TLS, it is necessary for the client to indicate the server to setup a TLS connection. After the client and server agree to use TLS, they will use a handshake process to negotiate a stateful connection.

A handshake begins with a client trying to connect to a TLS-enabled server, a client will request a secure connection and it will present a list of supported cipher suites including ciphers and hash functions. When the server end received the list, it will pick a cipher and hash function that it also supports, and then notify the client of the decision. Then, the server usually provides an identification in the form of a digital certificate, which contains the name of the server, the trusted certificate authority that guarantees the authenticity of the certificate, and the public encryption key of the server. On the client end, it will valid the certificate before proceeding. After the certificate has been confirmed, the client will generate session keys for the secure connection. There are two methods to generate these keys. First, the client encrypts a random number 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); The two parties then use random numbers to generate a unique session key to subsequently encrypt and decrypt the data during the session. Secondly, use Diffie-Hellman key exchange to securely generate a random and unique session key for encryption and decryption. This key has the additional attribute of forwarding confidentiality: if the server 's private key is disclosed in the future, it cannot be used to decrypt the current session, even if the session is intercepted and recorded by a third party.

This explains the handshake and the start of the secure connection, which uses the session key for encryption and decryption until the connection is closed. If any of the above steps fail, the TLS handshake fails and no connection will be created.

How TLS 1.2 protocol is used in Accuenergy products?

The communication module AXM-WEB2 supports the SSL encrypted web server plus the TSL 1.2 compliance for industry leading cybersecurity standard. With the standard TLS 1.2 connection, the communication module will be able to transmit the fast encrypted energy data to the server and only allow the access to owners and managers.