Formato de mensajes SSL/TLS

A continuación se detalla el formato de diversos mensajes del protocolo SSL/TLS, necesarios para interpretar la sesión SSL/TLS disponible para la práctica.

SSL Record Protocol

- Type (1): Subprotocolo que genera el mensaje enviado (20 para change cipher spec, 21 para alert, 22 para handshake y 23 para application).
- Version (2): Versión de SSL/TLS.
- Length (2): Longitud del fragmento enviado (payload) en bytes.

A partir de TLS 1.1, si el algoritmo de cifrado a usar implica un modo de operación CBC, para cada registro cifrado se indica explicitamente el valor IV antes del criptograma. La longitud de IV depende del algoritmo de cifrado negociado.

SSL Handshake protocol

Todos los mensajes empiezan con los mismos 2 campos:

- Type (1): Tipo de mensaje.
- Length (3): Longitud del contenido del mensaje en bytes.

A continuación se detallan los tipos de mensajes, indicando entre paréntesis el valor usado en el campo Type para cada tipo de mensaje.

ClientHello (1)

- Version (2): Versión más alta soportada por el cliente.
- Random (32): Valor aleatorio generado por el cliente.
- Session ID length (1): Longitud del Session ID que se quiere reanudar (0 si se quiere iniciar una nueva sesión).
- Session ID: El ID de la sesión que se quiere reanudar.
- Cipher suites length (2): Longitud de cipher suites propuestos (en bytes).
    - Cipher suites (2 por cipher suite): Códigos de cipher suites propuestos.
- Compresion length (1): Número de métodos de compresión propuestos.
    - Compresion methods (1 por método): Códigos de métodos de compresión propuestos.

A partir de TLS 1.2 (versiones previas deben ignorar los bytes a partir de aquí):

- Extensions length (2): Longitud de las extensiones en bytes.
    - Extension id (2 por extensión)
    - Extension length (2 por extension)
    - Extension content

ServerHello (2)

- Version (2): Versión SSL/TLS que se usará.
- Random (32): Valor aleatorio generado por el servidor.
- Session ID length (1): Longitud del Session ID (0 si no se quiere permitir reanudar sesiones).
- Session ID: El ID de la sesión (será la propuesta por el cliente si se acepta reanudarla o una nueva en caso contrario).
- Cipher suite (2): Código del cipher suite seleccionado.
- Compresion method (1): Código del método de compresión seleccionado.

A partir de TLS 1.2:

- Extensions length (2): Longitud de las extensiones en bytes.
  - Extension id (2 por extensión)
  - Extension length (2 por extension)
  - Extension content

Certificate (11)

- Certificate chain length (3): Longitud de la cadena de certificados en bytes.
- Certificate length (3 por cada certificado): Longitud del siguiente certificado de la cadena
- Certificate: Siguiente certificado de la cadena

ServerKeyExchange (12)

En caso de usar DHE key exchange:

- p length (2): Longitud del parámetro p (módulo) en bytes.
- p: Valor del parámetro p.
- g length (2): Longitud del parámetro g (generador) en bytes.
- g: Valor del parámetro g.
- Key length: Longitud de la clave pública en bytes.
- Key: Clave pública.

En caso de usar ECDHE key exchange:

- Curve type (1): Tipo de curva.
- Curve (2): Curva usada.
- Key length (1): Longitud de la clave pública.
- Key: Clave pública.

En ambos casos se añade una firma con la clave correspondiente al certificado del servidor. A partir de TLS 1.2 la firma se especifica usando los siguientes campos (hasta entonces dado que el algoritmo a usar era fijo la longitud de la firma también lo era y, por tanto, no se especificaban estos valores):

- Signature algorithm (2): Algoritmo hash y de firma usados para la firma del mensaje.

- Signature length (2): Longitud de la firma en bytes.
- Signature: Firma.

ServerHelloDone (14)

No tiene contenido.

ClientKeyExchange (16)

En caso de usar DH key exchange:

- Key length (2): Longitud de la clave pública en bytes.
- Key: Clave pública.

En caso de usar RSA key transport:

- Premaster secret: Premaster secret cifrado con la clave pública del certificado del servidor.

Finished (20)

Es el primer mensaje cifrado (al ir cifrado no podréis ver su estructura).

SSL Cipher Spec Protocol

Contiene un único mensaje con un único byte de contenido que siempre tiene el valor 1. Aunque se envía en el saludo, técnicamente no es parte del subprotocolo SSL Handshake.