Cómo funciona HTTPS

Déjame explicar cómo funciona HTTPS, incluyendo SSL, certificados, el proceso de cifrado y las diferencias entre TLS 1.2 y TLS 1.3, de una manera clara y paso a paso.

¿Qué es HTTPS?

HTTPS significa **Hypertext Transfer Protocol Secure**. Es la versión segura de HTTP, el protocolo utilizado para transferir datos entre tu navegador y un sitio web. La parte "segura"proviene de agregar una capa de cifrado y autenticación, proporcionada por **SSL (Secure Sockets Layer)** o su reemplazo moderno, **TLS (Transport Layer Security)**. Por lo tanto, HTTPS es esencialmente HTTP funcionando sobre TLS (o históricamente SSL) para mantener tus datos seguros.

¿Cómo Funciona HTTPS? El Apretón de Manos TLS

Cuando visitas un sitio web utilizando HTTPS (por ejemplo, `https://example.com`), tu navegador y el servidor del sitio web establecen una conexión segura antes de intercambiar cualquier dato. Este proceso se llama **apretón de manos TLS**. Aquí está cómo funciona en pasos simples:

1. **Saludo del Cliente**:
    - Tu navegador envía un mensaje al servidor diciendo, "¡Hola! Aquí está la versión de TLS que soporto (por ejemplo, TLS 1.3), los algoritmos de cifrado (suites de cifrado) que puedo usar, y una cadena aleatoria de bytes (aleatorio del cliente)."
2. **Saludo del Servidor**:
    - El servidor responde, "¡Hola de vuelta! Usaré esta versión de TLS y esta suite de cifrado de tu lista. Aquí está mi cadena aleatoria (aleatorio del servidor)."
3. **Certificado**:
    - El servidor envía su **certificado SSL**, que incluye su **clave pública** y está firmado por una **Autoridad Certificadora (CA) de confianza**. Este certificado prueba la identidad del servidor.
4. **Intercambio de Claves del Cliente**:
    - Tu navegador verifica el certificado para asegurarse de que es válido y firmado por una CA de confianza. Si pasa, el navegador genera un **secreto pre-maestro**, lo cifra con la clave pública del servidor y lo envía al servidor.
5. **Claves de Sesión**:

- Tanto el navegador como el servidor utilizan el aleatorio del cliente, el aleatorio del servidor y el secreto pre-maestro para generar independientemente la misma **clave de sesión**. Esta clave se utiliza para cifrar y descifrar todos los datos durante la sesión.

6. **Finalizado**:
    - Ambos lados envían un mensaje “finalizado”, cifrado con la clave de sesión, para confirmar que la conexión segura está lista.

Una vez completado el apretón de manos, todos los datos (como páginas web, formularios o archivos) se cifran con la clave de sesión, haciendo que sean ilegibles para cualquiera que pueda interceptarlos.

¿Qué Son los Certificados SSL y Cómo Funcionan?

Un **certificado SSL** es un documento digital que prueba la identidad de un sitio web y habilita el cifrado. Aquí está lo que necesitas saber:

- **Contenidos**: El certificado incluye el nombre de dominio del sitio web, su clave pública y una firma digital de una **Autoridad Certificadora (CA)**.
- **Propósito**: Asegura que el servidor es legítimo (por ejemplo, realmente te estás conectando a `example.com`, no a un sitio falso) y proporciona la clave pública para el cifrado.
- **Verificación**: Tu navegador verifica:
    1. ¿Es el certificado válido (no expirado o revocado)?
    2. ¿Está firmado por una CA de confianza? (Los navegadores tienen una lista incorporada de CAs de confianza, como DigiCert o Let’s Encrypt.)
- Si las verificaciones pasan, el navegador confía en el servidor y procede con el apretón de manos.

La CA actúa como un tercero de confianza que avala el sitio web. Sin esto, los atacantes podrían hacerse pasar por cualquier sitio y robar tus datos.

El Algoritmo de Cifrado

El cifrado en HTTPS se basa en una combinación de **criptografía asimétrica** y **criptografía simétrica**:

1. **Criptografía Asimétrica** (durante el apretón de manos):
    - Utiliza una **clave pública** (compartida abiertamente) y una **clave privada** (mantenida en secreto por el servidor).

- El navegador cifra el secreto pre-maestro con la clave pública del servidor. Solo el servidor, con su clave privada, puede descifrarlo.
- Ejemplos de algoritmos: RSA o Criptografía de Curva Elíptica (ECC).

2. **Criptografía Simétrica** (para la sesión):
   - Una vez creada la clave de sesión, ambos lados la utilizan para cifrar y descifrar datos.
   - Esto es más rápido que el cifrado asimétrico y es ideal para transferencias de datos grandes.
   - Ejemplo de algoritmo: AES (Advanced Encryption Standard).

El apretón de manos utiliza cifrado asimétrico para compartir la clave de sesión de manera segura, luego el cifrado simétrico toma el relevo para mayor eficiencia.

Diferencias Entre TLS 1.2 y TLS 1.3

TLS 1.2 y **TLS 1.3** son versiones del protocolo TLS, con TLS 1.3 siendo la versión más nueva y mejorada. Aquí están las diferencias clave:

Característica	TLS 1.2	TLS 1.3
Velocidad del Apretón de Manos	Varios viajes de ida y vuelta entre el cliente y el servidor, añadiendo latencia.	Menos viajes de ida y vuelta, a menudo solo uno, para conexiones más rápidas. Puede enviar datos inmediatamente (“0-RTT”) en reconexiones.
Suites de Cifrado	Soporta algoritmos de cifrado más antiguos y menos seguros (por ejemplo, SHA-1).	Elimina suites de cifrado obsoletas, solo usa las modernas y seguras (por ejemplo, AES-GCM).
Privacidad	Algunos datos del apretón de manos (como certificados) se envían en texto plano.	Casi todo el apretón de manos está cifrado, ocultando más detalles de los oyentes.
Resumen 0-RTT	No disponible.	Permite el envío instantáneo de datos al reconectar, pero corre el riesgo de ataques de repetición si no se asegura adecuadamente.
Seguridad	Vulnerable a ciertos ataques debido a características más antiguas.	Más fuerte por diseño, eliminando opciones débiles.

En resumen, TLS 1.3 es **más rápido**, **más seguro** y **más privado** que TLS 1.2, haciendo que sea el estándar para HTTPS moderno.

Juntando Todo

HTTPS asegura la comunicación web mediante: 1. Usar el **apretón de manos TLS** para establecer una conexión cifrada. 2. Confiar en **certificados SSL** de CAs de confianza para verificar la identidad del servidor. 3. Combinar **criptografía asimétrica** (para el apretón de manos) y **criptografía simétrica** (para la transferencia de datos) para proteger tus datos. 4. Evolucionar con protocolos como **TLS 1.3**, que mejora TLS 1.2 con mejor velocidad, seguridad y privacidad.

¡Ese es el núcleo de cómo HTTPS mantiene tu navegación segura! Hazme saber si te gustaría que profundizara en alguna parte.