DESPLIEGUE DE APLICACIONES WEB UD-1

HTTPS. SSH. CONEXIÓN A SERVIDOR

ÍNDICE

[¿QUÉ ES UN VPS? 3](#_Toc150542093)

[CONEXIÓN MEDIANTE SSH 4](#_Toc150542094)

[A. AUTENTICACIÓN 5](#_Toc150542095)

[1. Cifrados simétricos o de clave privada 5](#_Toc150542096)

[2. Cifrados asimétricos o de clave pública 6](#_Toc150542097)

[FASES DE CONEXIÓN EN UN SERVIDOR HTTPS SEGURO 7](#_Toc150542098)

[*A.* FASE 1 O *TCP HANDSHAKE* 8](#_Toc150542099)

[*B.* FASE 2 O *CERTIFICATE CHECK* (CIFRADO ASIMÉTRICO) 8](#_Toc150542100)

[*C.* FASE 3 O *KEY EXCHANGE* (CIFRADO ASIMÉTRICO) 10](#_Toc150542101)

[D. FASE 4 O DATA TRANSMISION (CIFRADO SIMÉTRICO) 10](#_Toc150542102)

# ¿QUÉ ES UN VPS?

Un servidor es una computadora en la que tu proveedor de alojamiento web almacena los archivos y las bases de datos necesarios para tu sitio web. Cada vez que un visitante en línea quiere acceder a tu sitio web, su navegador le envía una solicitud a tu servidor y transfiere los archivos necesarios a través de Internet. El alojamiento VPS te proporciona un servidor en la nube que simula un servidor físico; sin embargo, en realidad, la máquina se comparte entre varios usuarios.

Al usar la tecnología de virtualización, tu proveedor de alojamiento web instala una capa virtual sobre el sistema operativo del servidor. Esta capa divide el servidor en particiones y le permite a cada usuario instalar su propio sistema operativo y software.

Por lo tanto, un servidor privado virtual (VPS) es tanto virtual como privado porque tienes control absoluto. Está separado de otros usuarios del servidor a nivel del sistema operativo. De hecho, la tecnología VPS es similar a la creación de particiones en tu computadora cuando quieres ejecutar más de un sistema operativo (por ejemplo, Windows y Linux) sin tener que reiniciar.

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

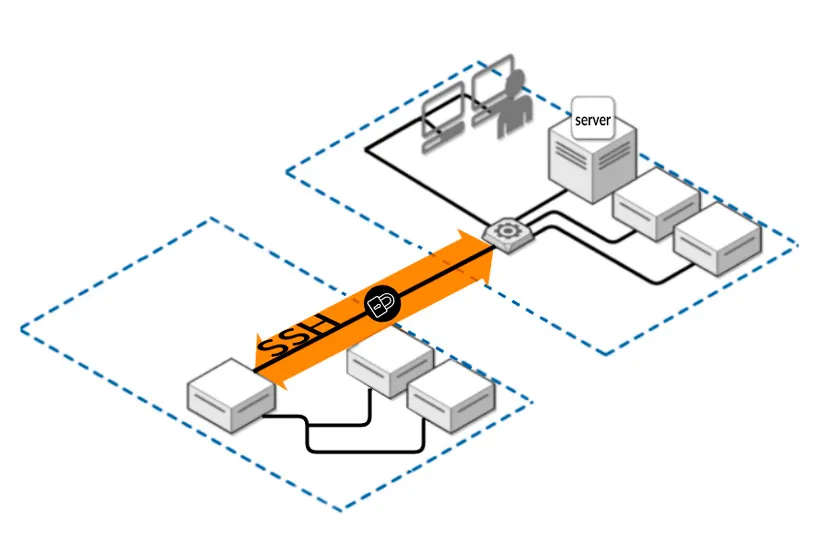
Un VPS te permite configurar tu sitio web dentro de un contenedor seguro con recursos garantizados (memoria, espacio en disco, núcleos de CPU, etc.) que no tienes que compartir con otros usuarios. Con el hosting VPS, tienes el mismo acceso de nivel raíz que si alquilaras un servidor dedicado, pero a un costo mucho más bajo.

El VPS es una solución más segura y estable que el hosting compartido, con el que no obtienes espacio de servidor dedicado. Sin embargo, es de menor escala y más barato que alquilar un servidor completo.

El hosting VPS generalmente es elegido por los propietarios de sitios web que tienen un tráfico de nivel medio que excede los límites de los planes de hosting compartido pero que aún no necesitan los recursos de un servidor dedicado.

# CONEXIÓN MEDIANTE SSH

Aunque nuestra máquina virtual esté en nuestro ordenador, ya hemos dicho que estamos simulando un VPS remoto. Para conectarnos a una máquina de forma remota y segura, la opción más recomendable es SSH.



SSH o Secure Shell es un protocolo de red criptográfico para operar servicios de red de forma segura a través de una red no protegida. Las aplicaciones típicas incluyen línea de comandos remota, inicio de sesión y ejecución de comandos remota, pero cualquier servicio de red puede protegerse con SSH.

SSH proporciona un canal seguro a través de una red no segura mediante el uso de una arquitectura cliente-servidor, conectando una aplicación cliente SSH con un servidor SSH. El puerto TCP estándar para SSH es 22 y se usa generalmente para acceder a sistemas operativos similares a Unix, pero también se puede usar en Microsoft Windows.

Proporciona un mecanismo para autenticar un usuario remoto, transferir entradas desde el cliente al host y retransmitir la salida de vuelta al cliente.

SSH tiene muchas aplicaciones diferentes:

* Gestión de servidores a los que no se puede acceder localmente
* Transferencia segura de archivos
* Creación de copias de seguridad
* Conexión entre dos ordenadores con encriptación de extremo a extremo
* Mantenimiento remoto desde otros ordenadores

## AUTENTICACIÓN

Los dos métodos de autenticación de usuario SSH más comunes que se utilizan son las contraseñas (cifrado simétrico) y las claves SSH (cifrado asimétrico o de clave pública). Los clientes envían contraseñas cifradas al servidor de forma segura. Sin embargo, las contraseñas son un método de autenticación arriesgado porque su solidez depende de que el usuario sepa qué hace que una contraseña sea segura.

Los pares de claves pública-privada SSH encriptados asimétricamente son una mejor opción. Una vez que el cliente descifra el mensaje, el servidor le otorga acceso al sistema.

Es decir, SSH opta por el cifrado híbrido, donde se utiliza el cifrado asimétrico para intercambiar unas claves que serán las que se utilizarán posteriormente en el intercambio de información.

1. Cifrados simétricos o de clave privada

Este tipo de cifrado utiliza la misma clave para cifrar y para descifrar la información. Por este motivo, la clave debe ser secreta y sólo conocida por el emisor y el receptor del mensaje.

Diagrama

Descripción generada automáticamente

**Ventajas**

* Muy rápidos → cifrar y descifrar un mensaje cada vez requiere un cierto tiempo, que si el algoritmo es complejo, puede ser elevado.

**Inconvenientes**

* Si alguien no autorizado consigue la clave, podrá espiar la comunicación sin problemas
* ¿Cómo hacemos para que emisor y receptar conozcan la clave en un primer momento? → no se puede transmitir por el canal inseguro → hay que transmitirla por otro canal seguro Ejemplos: PIN de la tarjeta del banco o archivo comprimido con contraseña

1. Cifrados asimétricos o de clave pública

En este tipo de cifrados cada usuario utiliza un par de claves: una clave pública y una clave privada. Un mensaje cifrado con la clave pública sólo se puede descifrar con su correspondiente clave privada y viceversa.

Diagrama

Descripción generada automáticamente

La clave pública es accesible a cualquier persona que quiera consultarla, no hace falta que sea transmitida por un canal seguro como en el caso anterior.

La clave privada sólo la debe conocer su dueño.

**Funcionamiento**

1. El emisor cifra un mensaje con la clave pública del receptor.
2. El receptor recibe el mensaje y es el único que podrá descifrarlo porque es el único que posee la clave cifrada asociada.

**Ventajas**

* No se necesita un nuevo canal independiente y seguro para transmitir la clave.

**Inconvenientes**

* Son más lentos que los cifrados simétricos.
* Hay que proteger muy bien la clave privada y tenerla siempre disponible para poder descifrar los mensajes (no es una contraseña).
* Hay que asegurarse de que la clave pública es de quién dice ser y no de un impostor que se esté haciendo pasar por él.

# FASES DE CONEXIÓN EN UN SERVIDOR HTTPS SEGURO

HTTPS funciona sobre TLS (aporta la capa de seguridad y cifrado.) TCP es el protocolo que envía y recibe datos.

Gráfico

Descripción generada automáticamente

Se va a trabajar entre TCP/UDP y el HTTP, es decir, en el TLS o SSL.

Un ejemplo de HTTP/3 sería por ejemplo la web de Google y un ejemplo de HTTP/2 sería el del correo web de educamadrid, en el caso de Google se trabaja bajo UDP, es decir, no se comprueba si la conexión se ha establecido y ha habido recepción de paquetes, sino que se envía la info de forma ininterrumpida, en el caso de educadamadrid sí que existe un protocolo TCP que necesita de una confirmación de la recepción de los paquetes.

En Google:

Tabla

Descripción generada automáticamente

En educaMadrid:

Imagen que contiene Tabla

Descripción generada automáticamente

Diagrama

Descripción generada automáticamente

## FASE 1 O *TCP HANDSHAKE*

En esta fase servidor y cliente se conectan y comunican de forma física, por TCP. Primero se envía un mensaje de sincronización (SYN del inglés *synchronize*) desde el cliente al servidor, a continuación, el servidor manda un mensaje que enlaza una sincronización con un “acuse de recibo” (ACK del inglés *acknowledgment)* y, por último, el cliente manda otro ACK al servidor.

## FASE 2 O *CERTIFICATE CHECK* (CIFRADO ASIMÉTRICO)

En esta fase el cliente va a comprobar que el servidor es quien dice ser. Para ello lo primero que se realiza es un envío cliente-servidor (client hello) seguido de un envío servidor-cliente (server hello) que lleva aparejada la llave pública. Una vez que la llave llegue al cliente, este a través de los certificados que tiene asociados, comprueba que dicha clave es real.

En el caso de que todo esté OK, en el navegador aparece el candado a la izquierda de la URL informando al usuario de que está en un sitio seguro.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

Descripción generada automáticamente

Si queremos comprobar el certificado SSL/TLS del sitio lo podemos hacer clicando en el candado, eligiendo la opción “La conexión es seguro” y, por último, haciendo clic en “El certificado es válido”.

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamenteInterfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

Este certificado está compuesto de una clave pública y otra privada. La clave privada es un fichero instalado y gestionado en el servidor web (o en algún dispositivo intermedio como un cortafuegos de aplicaciones web). La clave pública es lo que ha leído nuestro navegador para enseñarnos esta información:

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Además, si queremos comprobar los certificados que tenemos en el navegador lo podemos hacer accediendo a la configuración y buscando certificados.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

## FASE 3 O *KEY EXCHANGE* (CIFRADO ASIMÉTRICO)

En esta fase se va a llegar a cabo el intercambio de llaves cliente-servidor. Para ello lo primero que se va a realizar es la generación de una llave simétrica (*session key)* por parte del cliente a partir de la llave pública del servidor y la va a encriptar generando una *encrypted session key*.

Una vez que la *encrypted session key*, llega al servidor, este va a desencriptarla utilizando su clave privada (generando la *session key* original*)*. Aquí se está utilizando un cifrado asimétrico en todo momento.

Cuando esto se ha completado tanto cliente como servidor tienen una *session key* que les va a servir para empezar a transmitirse datos (webs, completos, etc.) mediante TCP haciendo uso de cifrado simétrico.

## FASE 4 O DATA TRANSMISION (CIFRADO SIMÉTRICO)

A partir de este momento comienza la transmisión de datos encriptados de forma simétrica tanto desde cliente-servidor como desde servidor-cliente.

Todas estas fases podéis verlo mediante esta captura realizada con la aplicación Wireshark de la conversación entre mi navegador Chrome y el servidor de correoweb.educa.madrid.org:

Texto

Descripción generada automáticamente

Para ver los ficheros intercambiados y alguna información interesante de aplicación cliente y servidora, abriendo en el navegador Chrome:

Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente

En otra comunicación, por ejemplo, aparece algo así eligiendo Network, el primer fichero de la izquierda y después la pestaña Headers, aparecen las cabeceras del protocolo HTTPS en la comunicación entre el cliente y el servidor:

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente