SSH

Es un protocolo (y programa) que permite acceder a un servidor remoto de una forma segura cifrando la información que se intercambia con el cliente.

La función mas usada es ejecutar comandos mediante terminar en una máquina remota, pero puede usarse para copiar (scp), ftp (sftp) y para crear "túneles" seguros usados pos multitud de aplicaciones.

Para cifrar la comunicación utiliza claves públicas y privadas.

El programa clientes se configura en el fichero `/etc/ssh/ssh_config` y el servidor en `/etc/ssh/sshd_config`.

Podemos iniciar sesión con el usuario y la contraseña o usando unas claves instaladas en el cliente y el servidor.

Directivas destacadas de /etc/ssh/sshd_config.

- **Port** : Número de puerto por el que escucha el servidor (22)
- **PermitRootLogin**: Indica si el root puede acceder mediante ssh o no
- **X11Forwarding**: Se permite el túnel para ejecutar programas de forma remota usando el entorno gráfico

Ficheros de claves privadas (según su sistema de cifrado)

CLIENTE	SERVIDOR
`~/.ssh/id_rsa`	`/etc/ssh/ssh_host_rsa_key`
`~/.ssh/id_dsa`	`/etc/ssh/ssh_host_dsa_key`
`~/.ssh/id_ecdsa`	`/etc/ssh/ssh_host_ecdsa_key`
`~/.ssh/id_ed25519`	`/etc/ssh/ssh_host_ed25519_key`

Las ficheros de clave pública son iguales pero acabados en `.pub`

ssh-keygen

Genera un par de claves publica y privada. Por defecto usa RSA. Opciones:

- **-t**: método de cifrado (rsa, dsa, ecdsa, ed25519, ...)
- **-b**: bits usados para el cifrado

```
ssh-keygen -t ecdsa -b 2084
```

Por defecto usará el directorio `~/.ssh/` para guardar las claves

Para acceder a un servidor con estas claves hay que guardar la clave pública en `~/.ssh/authorized_keys`

ssh-agent

Gestiona las claves privadas del usuario mientras que dure la sesión. Útil cuando se trabaja con varios servidores y se quiere agilizar la repetitiva identificación en cada uno de ellos.

Primero tenemos que ejecutar el comando para iniciarlo y después utilizar `ssh-add` para añadir todas las claves que queramos gestionar. Si hemos cambiado la ruta por defecto de las claves se la tendremos que indicar como parámetro.

Tunel SSH

Es una conexión cifrada entre dos puntos que se establece con la intención de que sea utilizada para trasmitir los datos de otra aplicación o servicio de forma segura.

```
ssh -N -f -L 8080:destino:80 usuario@origen
```

- `-N` : no ejecuta un comando
- `-f` : Se procesa en segundo plano
- `-L` : Especifica los puertos en el origen y el destino

Recomendaciones de seguridad para el servidor ssh

- **Protocol**: 2. No se puede admitir la versión 1
- **Port :** Es muy recomendable cambiar el puerto por defecto (22) por otro que no sea parecido (2222)
- **PermitRootLogin**: Se debe denegar el acceso a root, es mejor entrar como otro usuario y después subir de privilegios con su o sudo.
- **X11Forwarding**: Desativar, valor no.
- **PasswordAuthentication**: No. Se recomienda la autenticación por clave pública-privada.
- Desactivar **SSH tuneling.**
  - `AllowTcpForwarding no`
  - `AllowStreamLocalForwarding no`
  - `GatewayPorts no`
  - `PermitTunnel no`

Otras directivas a tener en cuenta

- **MaxAuthTries**: Número de intentos de login antes de cortar la conexión.
- **AllowGroups / AllowUsers**: Permitir sólo unos grupos o usuarios determinados
- **PrintLastLog**: no. Desactiva la información del último login
- **PrintMotd** no. Desactiva el mensaje de entrada (/etc/motd).

Estas dos directivas NO desactivan el mensaje de bienvenida del propio shell, para ello se puede usar la directiva HUSHLOGIN_FILE del fichero /etc/login.defs o añadir un fichero al home del usuario `touch ~/.hushlogin.`
También se deberían revisar las opciones de motd en /etc/pam.d/sshd

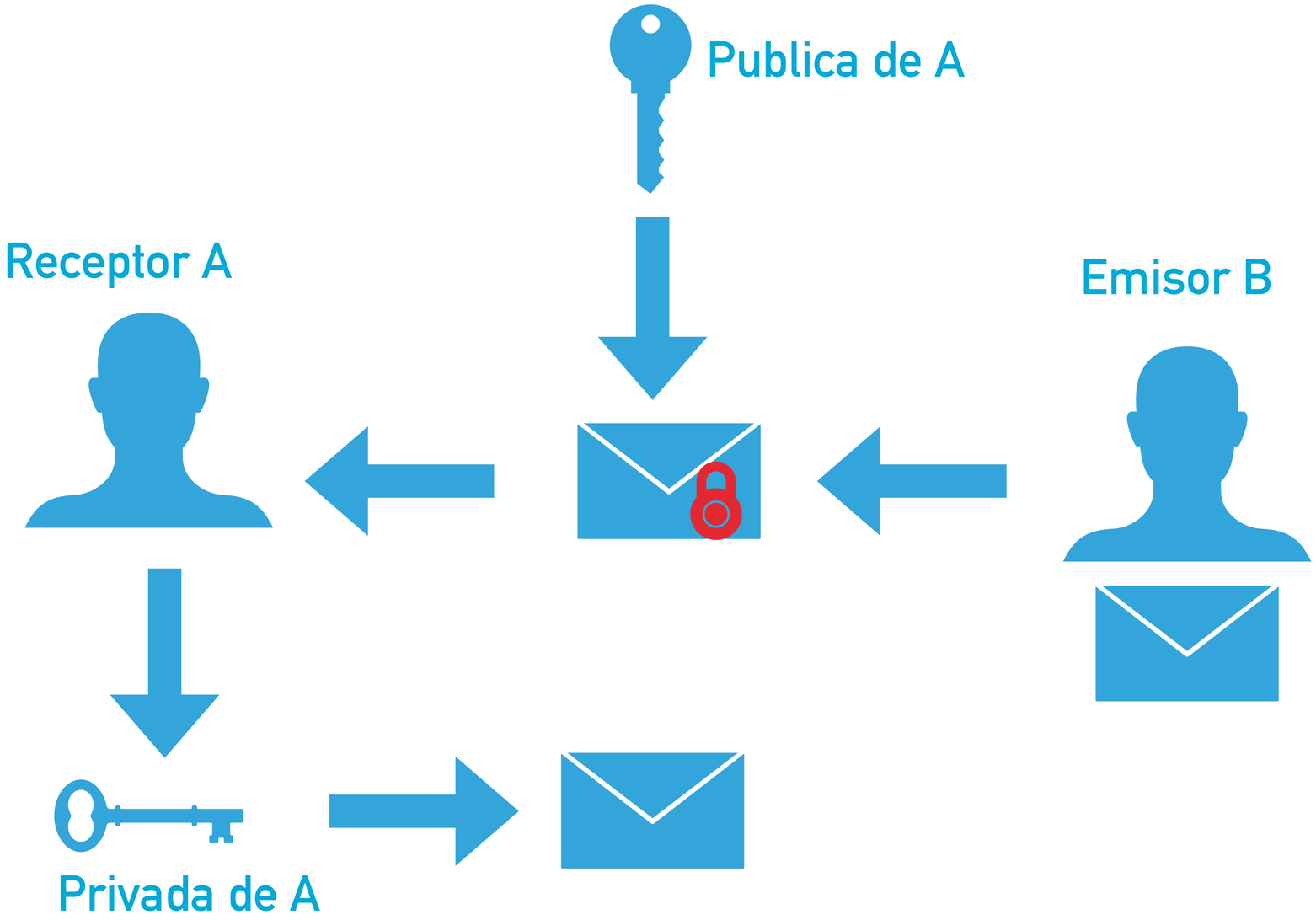
GPG

Sirve para cifrar un fichero usando claves asimétricas. También puede firmar digitalmente un texto, para que el mensaje y el remitente pueden ser verificados.

Comandos

- `gpg --gen-key` : Genera la clave pública y privada
- `gpg --output pub_key_file --export 950B76C6` : Exporta la clave pública `950B76C6` al fichero `pub_key_file`
- `gpg --import pub_key_file` : Importa la clave pública `pub_key_file`
- `gpg --encrypt --recipient 83726383 mensaje.txt` : Cifra el mensaje
- `gpg [-d] mensaje.txt.gpg` : Descifra el mensaje

PROTECCIÓN DE DATOS MEDIANTE CIFRADO



PROTECCIÓN DE DATOS MEDIANTE CIFRADO