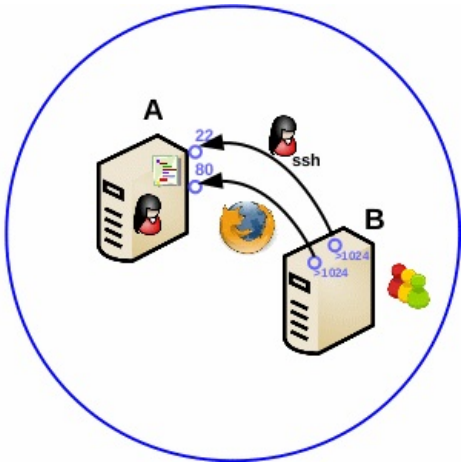
Conexión Remota mediante SSH
Cambios en sshd_config

ESCENARIO

Máquinas virtuais:
RAM ≤ 2048MB CPU ≤ 2 PAE/NX habilitado
Rede: 192.168.120.0

Máquina virtual A:	**Máquina virtual B:**
Rede Interna	Rede Interna
Servidor SSH: openssh-server	Cliente SSH: openssh-client (ssh)
Servidor Web: Apache (apache2)	Cliente Web: Navegador (firefox)
ISO: Kali Live amd64	ISO: Kali Live amd64
IP/MS: 192.168.120.100/24	IP/MS: 192.168.120.101/24

BIOS: Permite arranque dispositivo extraíble: CD/DVD, USB



NOTAS:

- **Servidor ssh GNU/Linux**:
  - Paquete openssh-server (# apt update && apt -y install openssh-server).
  - Ficheiro de configuración: **/etc/ssh/sshd_config (man sshd_config)**

    PermitRootLogin ➜ Directiva que determina se o usuario root pode acceder a conectarse mediante SSH.

    Port ➜ Directiva que determina o porto TCP de escoita para o servidor SSH. Poden existir múltiples liñas Port indicando diferentes portos de escoita do servizo SSH.

    ListenAddress ➜ Directiva que determina que direccións locais escoitan no servizo SSH. Poden existir múltiples liñas ListenAddress indicando varias interfaces de escoita do servizo SSH.

    X11Forwarding ➜ Directiva que determina se a redirección gráfica é posible mediante conexións SSH. Pode soamente tomar 2 valores: yes/no.

    X11DisplayOffset ➜ Directiva que determina o número do display onde espera o servidor gráfico. Por defecto é 10, para evitar interferencias con reais servidores X11.

    X11UseLocalhost ➜ Directiva que determina se a redirección gráfica é posible na dirección loopback ou en calquera dirección.

    PubKeyAuthentication ➜ Directiva que determina se a autenticación de cifrado asimétrico (ou cifrado de clave pública) está permitida. Por defecto está permitida tomando o valor *yes*

    AuthorizedKeysFile ➜ Directiva que especifica o ficheiro que contén as claves públicas empregadas para a autenticación de usuario. Por defecto o ficheiro toma o valor *~/.ssh/authorized_keys* ou *~/.ssh/authorized_keys2*

    PermitEmptyPasswords ➜ Directiva que especifica se a autenticación de usuario é posible con contrasinais baleiras. Por defecto, toma o valor *no* impedindo o acceso cunha contrasinal baleira.

    PasswordAuthentication ➜ Directiva que especifica se a autenticación de usuario é posible mediante contrasinal. Por defecto, toma o valor *yes* permitindo o acceso a través de contrasinal. Soe configurarse a *no* cando soamente interesa acceder mediante cifrado asimétrico.

    MaxAuthTries ➜ Directiva que especifica o número máximo de reintentos de autenticación por conexión. Por defecto son *6*

NOTAS:

- **Servidor gráfico X (Xorg)**: Variable de contorna **DISPLAY (man X ; man ssh)**

  Un **display** consta mínimo de 3 elementos: teclado, rato e pantalla. Dende a perspectiva dun usuario todo servidor gráfico ten un display, o cal defínese como ➜ hostname:displaynumber.screennumber

  DISPLAY=hostname:displaynumber.screennumber ➜ Variable de contorna que permite definir o display dun servidor gráfico:

  - hostname: Especifica o nome do host no cal o display está fisicamente conectado. Se non se define enténdese que a comunicación ao servidor gráfico sobre a mesma máquina (maquina local) terá lugar da forma máis eficiente.
  - displaynumber: É o único valor do DISPLAY que sempre debe configurarse. O termo display é usado normalmente para referirse a un conxunto de monitores que comparten un conxunto común de dispositivos de entrada (teclado, rato, tablet, etc.). A maioría dos equipos soamente posúen un display, pero pode ser que posúan varíos. xa que é necesario que varios usuarios poidan traballar nunha contorna gráfica simultaneamente. Para evitar confusión, cada display sobre un equipo ten asignado un número de display (comenzando en 0) cando o servidor gráfico (X) arranca para ese display. **O número do display sempre debe darse para configurar o DISPLAY**
  - screennumber: Algúns displays comparten os seus dispositivos de entrada con 2 ou máis monitores, que poden ser configurados como unha única pantalla lóxica, o al permite ás ventás moverse entre as pantallas, ou como pantallas individuais. Se se configura como que cada monitor ten as súas propias ventás, cada pantalla é asignada a un screen number. Se non se especifica o screen number toma o valor 0

  DISPLAY=:0.0 ➜ Display por defecto, que equivale a hostname=comunicación máis eficiente co servidor gráfico na mesma máquina local, displaynumber=0 e screennumber=0

  ```
  $ declare -p | grep DISPLAY
  declare -x DISPLAY=":0.0" #Amosa o valor da variable DISPLAY e como está declarada
  ```
  . Vemos que a variable está exportada, de tal xeito que a variable é válida na contorna actual da shell e en calquera subshell.

  Ben, pero que acontece cando non conectamos dende a máquina local senón dende a rede? Pois necesitamos definir o DISPLAY para que poida ser empregado dende a rede. Por exemplo, exportando a variable co nome do noso hostname. Así, se o hostname posúe o valor kaliA deberiamos facer o seguinte:

  ```
  $ declare -x DISPLAY=kaliA:0
  ```

  Nas conexións SSH se empregamos o comando ssh coa opción -X podemos executar un programa no servidor gráfico remoto e visualizalo no equipo cliente. Neste caso, automaticamente xa o comando ssh configura a variable DISPLAY correctamente para poder facer a redirección gráfica.

  A maioría dos programas aceptan na liña de comandos a opción **-display DISPLAY** ou **--display DISPLAY**, a cal **sobreescribe** o contido da variable **DISPLAY**:

  ```
  $ xeyes -display :0.0 & #Executar en segundo plano o comando xeyes no display :0.0 (hostname=servidor gráfico local, display=0, screennumber=0)
  ```

  Podemos arrancar outro sistema X Window (servidor gráfico + cliente gráfico) mediante o comando **startx (man startx; man xinit)**:

  ```
  $ su - -c "startx -- :10" #No comando startx antes dos caracteres -- teñen lugar as opcións do cliente gráfico, e logo teñen lugar as opcións do servidor gráfico. Así, creamos o display :10 (hostname=servidor gráfico local, display=10, screennumber=0). O comando é executado mediante o usuario root a través de su - -c
  $ su - -c "xeyes -display :10" #Executar coma root o comando xeyes no display :10 (hostname=servidor gráfico local, display=10, screennumber=0)
  ```

Práctica - Conexión Remota mediante SSH - Cambios en sshd_config

Máquina virtual A: Kali amd64

1. Pór contrasinal ao usuario, co que arranca a live, de nome: kali
   Na contorna gráfica abrir un terminal e executar:

   kali@kali:~$ setxkbmap es #Cambiar o mapa de teclado ao idioma español.
   kali@kali:~$ passwd kali || (echo -e 'kali\nabc123.\nabc123.' | passwd) #Cambiar o contrasinal do usuario kali. Por como contrasinal **abc123.** (Ollo que o contrasinal ten un caracter punto final).
   kali@kali:~$ sudo passwd root || (sudo -c "echo -e 'abc123.\nabc123.' | passwd") #Cambiar o contrasinal do usuario root. Por como contrasinal **abc123.** (Ollo que o contrasinal ten un caracter punto final). O cambio de contrasinal é posible debido aos permisos configurados co comando sudo (/etc/sudoers, visudo).

2. Cambiar hostname da máquina virtual A. Por kaliA como hostname:
   kali@kali:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
   root@kali:~# echo 'kaliA' > /etc/hostname #Indicar ao sistema o valor do hostname.
   root@kali:~# echo 'kernel.hostname=kaliA' >> /etc/sysctl.conf #Indicar ao kernel o valor do hostname.
   root@kali:~# sysctl -p #Activar o cambio de hostname sen ter que pechar sesión nin reiniciar
   root@kali:~# exit #Saír da consola local sudo na que estabamos a traballar para voltar á consola local de **kali**.
   kali@kali:~$ exit #Pechar o terminal saíndo da consola local do usuario **kali**.

3. Configurar a rede:
   Na contorna gráfica abrir un terminal e executar:

   kali@kaliA:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
   root@kaliA:~# /etc/init.d/avahi-daemon stop #Parar o demo avahi-daemon(control resolución de nomes) para poder configurar de forma manual a configuración de rede e non ter conflicto con este demo.
   root@kaliA:~# /etc/init.d/network-manager stop #Parar o demo network-manager(xestor de rede) para poder configurar de forma manual a configuración de rede e non ter conflicto con este xestor.
   root@kaliA:~# ip addr show #Amosar a configuración de todas as tarxetas de rede. Nesta caso, na máquina A, as tarxetas de redes: loopback(lo) e interna(eth0).
   root@kaliA:~# ip addr add 192.168.120.100/24 dev eth0 #Configurar a tarxeta de rede interna eth0, coa IP: 192.168.120.100 e máscara de subrede: 255.255.255.0.
   root@kaliA:~# ip addr show #Amosar a configuración de todas as tarxetas de rede. Nesta caso, na máquina A, as tarxetas de redes: loopback(lo) e interna(eth0).
   root@kaliA:~# ping -c4 192.168.120.100 #Comprobar mediante o comando ping a conectividade coa interface de rede local eth0

4. Comprobar estado do Servidor SSH:
   root@kaliA:~# /etc/init.d/ssh status #Comprobar o estado do servidor SSH, por defecto non está arrancado.
   root@kaliA:~# nc -vz localhost 22 #Mediante o comando nc(netcat) comprobar se o porto 22 do servidor ssh está en estado escoita(listen), esperando conexións. A opción -v corresponde á opción verbose, o que permite amosar información máis detallada na saída do comando. A opción -z permite devolver PROMPT do sistema e de igual xeito facer o escaneo ao/s porto/s solicitados. O número 22 é o porto TCP a escanear.
   root@kaliA:~# nc -vz 192.168.120.100 22 #Mediante o comando nc(netcat) comprobar se o porto 22 do servidor ssh está en estado escoita(listen), esperando conexións. A opción -v corresponde á opción verbose, o que permite amosar información máis detallada na saída do comando. A opción -z permite devolver PROMPT do sistema e de igual xeito facer o escaneo ao/s porto/s solicitados. O número 22 é o porto TCP a escanear.
   root@kaliA:~# netstat -natp | grep 22 #Mediante o comando netstat comprobar que o porto 22 do servidor SSH está en estado escoita(listen), esperando conexións. A opción -n permite non resolver nomes amosando así soamente as IPs e o comando ser máis rápido na execución. A opción -a equivale á opción all o que permite amosar todos os sockets (conectores) á escoita no servidor. A opción -t equivale a tcp o que permite buscar soamente información sobre o protocolo TCP. A opción -p equivale a program e amosa o PID e nome do programa ao cal pertence o socket.
   root@kaliA:~# ss -natp | grep 22 #Mediante o comando ss comprobar que o porto 22 do servidor SSH está en estado escoita(listen), esperando conexións. A opción -n permite non resolver nomes amosando así soamente as IPs e o comando ser máis rápido na execución. A opción -a equivale á opción all o que permite amosar todos os sockets (conectores) á escoita no servidor. A opción -t equivale a tcp o que permite buscar soamente información sobre o protocolo TCP. A opción -p equivale a program e amosa o PID e nome do programa ao cal pertence o socket.
   root@kaliA:~# /etc/init.d/ssh start #Arrancar o servidor SSH.

root@kaliA:~# /etc/init.d/ssh status #Comprobar o estado do servidor SSH, agora debe estar arrancado.
root@kaliA:~# find /etc/rc* -name "*ssh*" #Busca polas links runlevels nos cartafoles /etc/rc*
root@kaliA:~# systemctl enable ssh #Permite que o servizo ssh sexa iniciado no arranque xerando os links nos runlevels (/etc/rcX.d)
root@kaliA:~# find /etc/rc* -name "*ssh*" #Busca polas links runlevels nos cartafoles /etc/rc*
root@kaliA:~# systemctl is-enabled ssh.service #Amosa se o servizo ssh está enabled ou disabled
root@kaliA:~# nc -vz 192.168.120.100 22 #Mediante o comando nc(netcat) comprobar se o porto 22 do servidor ssh está en estado escoita(listen), esperando conexións. A opción -v corresponde á opción verbose, o que permite amosar información máis detallada na saída do comando. A opción -z permite devolver PROMPT do sistema e de igual xeito facer o escaneo ao/s porto/s solicitados. O número 22 é o porto TCP a escanear.
root@kaliA:~# ssh -v kali@localhost #Comprobar se o servidor SSH está activo e podemos conectarnos a el dende localhost co usuario kali e o seu contrasinal. Se é a primeira ver que nos conectamos o servidor avísanos se estamos de acordo coa autenticación. Respostamos yes e pulsamos Enter. A opción -v (modo verbose) aporta información máis detallada da conexión.
root@kaliA:~# exit #Saír da consola remota ssh a que acabamos de acceder, para voltar á consola local de **root**.

5. sshd_config: Impedir/Permitir a root a conexión mediante SSH

PermitRootLogin ➜ Directiva que determina se o usuario root pode acceder a conectarse mediante SSH. Así, pode tomar os seguintes valores:

- PermitRootLogin prohibit-password ➜ o usuario root non poderá conectar mediante SSH.
- PermitRootLogin without-password ➜ igual que prohibit-password, pero está obsoleta (en desuso).
- PermitRootLogin yes ➜ o usuario root poderá conectar mediante SSH.
- PermitRootLogin forced-commands-only ➜ o usuario root poderá conectar realizar conexións de comandos (non consola) SSH mediante cifrado asimétrico (cifrado de clave pública).
- PermitRootLogin no ➜ deshabilita ao usuario root o acceso mediante conexión SSH.

kali@kaliA:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)

root@kaliA:~# grep -n '^#PermitRootLogin' /etc/ssh/sshd_config #Buscar no ficheiro /etc/ssh/sshd_config mediante o comando *grep* o patrón de texto '*#PermitRootLogin*' nun comezo de liña ^ indicando o número/s da/s liña/s (*-n*) onde existe o patrón buscado.

root@kaliA:~# VAR=$(grep -n '^#PermitRootLogin' /etc/ssh/sshd_config | cut -d ':' -f1 | tail -1) #Crear a variable VAR co contido o número da última liña (poden existir máis de 1 se o ficheiro xa fora modificado) do patrón buscado no ficheiro /etc/ssh/sshd_config

root@kaliA:~# sed -i "s/PermitRootLogin/#PermitRootLogin/g" /etc/ssh/sshd_config #Modificar no ficheiro /etc/ssh/sshd_config o patrón de texto *PermitRootLogin* por *#PermitRootLogin* en todas as concurrencias atopadas, inclusive que fosen na mesma liña, é dicir, comentar (mediante o caracter #) para que non teña lugar a directiva *PermitRootLogin*

root@kaliA:~# sed -i "${VAR}a\PermitRootLogin yes" /etc/ssh/sshd_config #Activar o acceso a root mediante conexión SSH, é dicir, modificar o ficheiro /etc/ssh/sshd_config engadindo a liña *PermitRootLogin yes* logo da liña atopada anteriormente que comeza por *#PermitRootLogin*

root@kaliA:~# /etc/init.d/ssh reload #Recargar o ficheiro de configuración do servidor SSH para que se activen os cambios realizados

root@kaliA:~# ssh -v root@localhost #Realizar unha conexión SSH a localhost mediante o usuario root e o seu contrasinal. Se é a primeira ver que nos conectamos o servidor avísanos se estamos de acordo coa autenticación. Respostamos yes e pulsamos Enter. A opción -v (modo verbose) aporta información máis detallada da conexión.

root@kaliA:~# exit #Saír da consola remota ssh a que acabamos de acceder, para voltar á consola local de **root**.

root@kaliA:~# sed -i 's/PermitRootLogin yes/PermitRootLogin prohibit-password/' /etc/ssh/sshd_config #Deshabilitar conexións SSH ao usuario root.

root@kaliA:~# /etc/init.d/ssh reload #Recargar o ficheiro de configuración do servidor SSH para que se activen os cambios realizados

root@kaliA:~# ssh -v root@localhost #Realizar unha conexión SSH a localhost mediante o usuario root e o seu contrasinal. Se é a primeira ver que nos conectamos o servidor avísanos se estamos de acordo coa autenticación. Respostamos yes e pulsamos Enter. A opción -v (modo verbose) aporta información máis detallada da conexión.
Neste caso como modificamos a directiva *PermitRootLogin* a *prohibit-password* o acceso co usuario *root* non é posible.

root@kaliA:~# exit #Saír da consola local do usuario **root** para voltar á consola local do usuario **kali**.

6. sshd_config: Modificar o porto TCP de conexión SSH

Port ➜ Directiva que determina o porto TCP de escoita para o servidor SSH. Poden existir múltiples liñas Port indicando diferentes portos de escoita do servizo SSH:

- Port 22 ➜ por defecto o servidor SSH espera no porto TPC 22 (ver /etc/services).
- Port 4444 ➜ modificación do porto de escoita do servidor SSH ao porto TCP 4444.

kali@kaliA:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)

root@kaliA:~# grep -n '^#Port' /etc/ssh/sshd_config #Buscar no ficheiro /etc/ssh/sshd_config mediante o comando *grep* o patrón de texto '*#Port*' nun comezo de liña ^ indicando o número/s da/s liña/s (*-n*) onde existe o patrón buscado.

root@kaliA:~# VAR=$(grep -n '^#Port' /etc/ssh/sshd_config | cut -d ':' -f1 | tail -1) #Crear a variable VAR co contido o número da última liña (poden existir máis de 1 se o ficheiro xa fora modificado) do patrón buscado no ficheiro /etc/ssh/sshd_config

root@kaliA:~# sed -i "s/Port/#Port/g" /etc/ssh/sshd_config #Modificar no ficheiro /etc/ssh/sshd_config o patrón de texto *Port* por *#Port* en todas as concurrencias atopadas, inclusive que fosen na mesma liña, é dicir, comentar (mediante o caracter #) para que non teña lugar a directiva *Port*

root@kaliA:~# sed -i "${VAR}a\Port 4444" /etc/ssh/sshd_config #Activar o porto TCP de escoita 4444 para esperar conexións SSH, é dicir, modificar o ficheiro /etc/ssh/sshd_config engadindo a liña *Port 4444* logo da liña atopada anteriormente que comeza por *#Port*

root@kaliA:~# /etc/init.d/ssh reload #Recargar o ficheiro de configuración do servidor SSH para que se activen os cambios realizados

root@kaliA:~# ssh -p 4444 kali@localhost #Realizar unha conexión SSH a localhost mediante o usuario kali e o seu contrasinal a través do porto TCP 4444. Se é a primeira ver que nos conectamos o servidor avísanos se estamos de acordo coa autenticación. Respostamos yes e pulsamos Enter.

kali@kaliA:~$ exit #Saír da consola remota ssh a que acabamos de acceder, para voltar á consola local de **root**.

root@kaliA:~# sed -i "${VAR}a\Port 22" /etc/ssh/sshd_config #Activar o porto TCP de escoita 22 para esperar conexións SSH, é dicir, modificar o ficheiro /etc/ssh/sshd_config engadindo a liña *Port 22* logo da liña atopada anteriormente que comeza por *#Port*

root@kaliA:~# /etc/init.d/ssh reload #Recargar o ficheiro de configuración do servidor SSH para que se activen os cambios realizados

root@kaliA:~# ssh -p 22 kali@localhost #Realizar unha conexión SSH a localhost mediante o usuario kali e o seu contrasinal a través do porto TCP 22. Se é a primeira ver que nos conectamos o servidor avísanos se estamos de acordo coa autenticación. Respostamos yes e pulsamos Enter.

kali@kaliA:~$ exit #Saír da consola remota ssh a que acabamos de acceder, para voltar á consola local de **root**.

root@kaliA:~# exit #Saír da consola local do usuario **root** para voltar á consola local do usuario **kali**.

7. sshd_config: Especificar as interfaces de rede de escoita para o servizo SSH

ListenAddress ➜ Directiva que determina que direccións locais escoitan no servizo SSH. Poden existir múltiples liñas ListenAddress indicando varias interfaces de escoita do servizo SSH. Pódese empregar os seguintes formatos:

- ListenAddress 0.0.0.0 ➜ é a opción por defecto. O servizo SSH está a escoita de conexións en todas as interfaces de rede locais.
- ListenAddress localhost ➜ indica o hostname onde espera a escoita o servizo SSH, neste caso localhost (ver /etc/hosts)
- ListenAddress kaliA ➜ indica o hostname onde espera a escoita o servizo SSH, neste caso kaliA (ver /etc/hosts)
- ListenAddress kaliA:5555 ➜ indica o hostname e o porto TCP onde espera á escoita o servizo SSH, neste caso no host kaliA (ver /etc/hosts) e no porto TCP 5555
- ListenAddress 127.0.0.1 ➜ indica a IP onde espera o a escoita o servizo SSH, neste caso 127.0.0.1
- ListenAddress 127.0.0.1:6666 ➜ indica a IP e o porto TCP onde espera á escoita o servizo SSH, neste caso 127.0.0.1 e no porto TCP 6666

kali@kaliA:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)

root@kaliA:~# grep -n '^#ListenAddress' /etc/ssh/sshd_config #Buscar no ficheiro /etc/ssh/sshd_config mediante o comando *grep* o patrón de texto '*#ListenAddress*' nun comezo de liña ^ indicando o número/s da/s liña/s (*-n*) onde existe o patrón buscado.

root@kaliA:~# VAR=$(grep -n '^#ListenAddress' /etc/ssh/sshd_config | cut -d ':' -f1 | tail -1) #Crear a variable VAR co contido o número da última liña (poden existir máis de 1 se o ficheiro xa fora modificado) do patrón buscado no ficheiro /etc/ssh/sshd_config

root@kaliA:~# sed -i "s/ListenAddress/#ListenAddress/g" /etc/ssh/sshd_config #Modificar no ficheiro /etc/ssh/sshd_config o patrón de texto *ListenAddress* por *#ListenAddress* en todas as concurrencias atopadas, inclusive que fosen na mesma liña, é dicir, comentar (mediante o caracter #) para que non teña lugar a directiva *ListenAddress*

root@kaliA:~# sed -i "${VAR}a\ListenAddress 127.0.0.1" /etc/ssh/sshd_config #Pór soamente (xa que comentamos anteriormente todas as directivas ListenAddress) á escoita a IP interface loopback(lo) 127.0.0.1.

root@kaliA:~# /etc/init.d/ssh reload #Recargar o ficheiro de configuración do servidor SSH para que se activen os cambios realizados

root@kaliA:~# ssh -l kali 127.0.0.1 #Intento de conexión SSH a través do porto TCP 22 en 127.0.0.1 mediante o usuario kali e o seu contrasinal. Se é a primeira ver que nos conectamos o servidor avísanos se estamos de acordo coa autenticación. Respostamos yes e pulsamos Enter.

kali@kaliA:~$ exit #Saír da consola remota ssh a que acabamos de acceder, para voltar á consola local de **root**.

root@kaliA:~# ssh kali@localhost #Intento de conexión SSH a través do porto TCP 22 en localhost (/etc/hosts) mediante o usuario kali e o seu contrasinal. Se é a primeira ver que nos conectamos o servidor avísanos se estamos de acordo coa autenticación. Respostamos yes e pulsamos Enter.

kali@kaliA:~$ exit #Saír da consola remota ssh a que acabamos de acceder, para voltar á consola local de **root**.

root@kaliA:~# sed -i "${VAR}a\ListenAddress localhost:6666" /etc/ssh/sshd_config #Pór á escoita o host localhost (interface loopback(lo) 127.0.0.1) no porto TCP 6666.

root@kaliA:~# /etc/init.d/ssh reload #Recargar o ficheiro de configuración do servidor SSH para que se activen os cambios realizados

root@kaliA:~# ssh -p 6666 kali@localhost #Intento de conexión SSH a través do porto TCP 6666 en localhost mediante o usuario kali e o seu contrasinal. Se é a primeira ver que nos conectamos o servidor avísanos se estamos de acordo coa autenticación. Respostamos yes e pulsamos Enter.

kali@kaliA:~$ exit #Saír da consola remota ssh a que acabamos de acceder, para voltar á consola local de **root**.

root@kaliA:~# sed -i "${VAR}a\ListenAddress 192.168.120.100:7777" /etc/ssh/sshd_config #Pór á escoita a interface configurada coa IP 192.168.120.100 no porto TCP 7777.

root@kaliA:~# /etc/init.d/ssh reload #Recargar o ficheiro de configuración do servidor SSH para que se activen os cambios realizados

root@kaliA:~# ssh -p 7777 kali@localhost #Intento de conexión SSH a través do porto TCP 7777 en localhost mediante o usuario kali e o seu contrasinal. Pero como localhost non está configurado para acceder por ese porto non é posible a conexión.

root@kaliA:~# ssh -p 7777 -l kali 192.168.120.100 #Intento de conexión SSH a través do porto TCP 7777 en 192.168.120.100 mediante o usuario kali e o seu contrasinal. Se é a primeira ver que nos conectamos o servidor avísanos se estamos de acordo coa autenticación. Respostamos yes e pulsamos Enter.

kali@kaliA:~$ exit #Saír da consola remota ssh a que acabamos de acceder, para voltar á consola local de **root**.

root@kaliA:~# exit #Saír da consola local do usuario **root** para voltar á consola local do usuario **kali**.

8. **sshd_config: Redireccionar X (X11Forwarding, X11DisplayOffset, X11UseLocalhost) para realizar conexións gráficas remotas en conexións SSH**

X11Forwarding ➜ Directiva que determina se a redirección gráfica é posible mediante conexións SSH. Pode soamente tomar 2 valores: yes/no.

- X11Forwarding no ➜ é o valor por defecto. Deshabilita a redirección gráfica do servidor SSH.
- X11Forwarding yes ➜ Habilita a redirección gráfica do servidor SSH.

X11DisplayOffset ➜ Directiva que determina o número do display onde espera o servidor gráfico. Por defecto é 10, para evitar interferencias con reais servidores X11.

- X11DisplayOffset 10 ➜ é o valor por defecto. Indica o número de display onde espera o servidor gráfico para conexións SSH.
- X11DisplayOffset 100 ➜ Indica o número de display a 100 onde espera o servidor gráfico para conexións SSH.
- X11Forwarding yes ➜ Habilita a redirección gráfica do servidor SSH.

X11UseLocalhost ➜ Directiva que determina se a redirección gráfica é posible na dirección loopback ou en calquera dirección:

- X11UseLocalhost yes ➜ é o valor por defecto. Permite a redirección gráfica á dirección loopback e define a variable de entorno DISPLAY a localhost, o cal prevén conexións remotas non permitidas ao display.
- X11UseLocalhost no ➜ Habilita a redirección gráfica a todas as interfaces de rede.

kali@kaliA:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)

root@kaliA:~# grep -n '^X11Forwarding' /etc/ssh/sshd_config #Buscar no ficheiro /etc/ssh/sshd_config mediante o comando *grep* o patrón de texto '*X11Forwarding*' nun comezo de liña ^ indicando o número/s da/s liña/s (*-n*) onde existe o patrón buscado.

root@kaliA:~# VAR=$(grep -n '^X11Forwarding' /etc/ssh/sshd_config | cut -d ':' -f1 | tail -1) #Crear a variable VAR co contido o número da última liña (poden existir máis de 1 se o ficheiro xa fora modificado) do patrón buscado no ficheiro /etc/ssh/sshd_config

root@kaliA:~# sed -i "s/X11Forwarding/#X11Forwarding/g" /etc/ssh/sshd_config #Modificar no ficheiro /etc/ssh/sshd_config o patrón de texto *X11Forwarding* por *#X11Forwarding* en todas as concurrencias atopadas, inclusive que fosen na mesma liña, é dicir, comentar (mediante o caracter #) para que non teña lugar a directiva *X11Forwarding*

root@kaliA:~# sed -i "${VAR}a\X11Forwarding yes" /etc/ssh/sshd_config #Activar o acceso a root mediante conexión SSH, é dicir, modificar o ficheiro /etc/ssh/sshd_config engadindo a liña *X11Forwarding yes* logo da liña atopada anteriormente que comeza por *#X11Forwarding*

root@kaliA:~# /etc/init.d/ssh reload #Recargar o ficheiro de configuración do servidor SSH para que se activen os cambios realizados

root@kaliA:~# ssh -X kali@localhost #Realizar unha conexión SSH a localhost mediante o usuario kali e o seu contrasinal a través do porto TCP 22 solicitando a redirección gráfica (*-X*). Se é a primeira ver que nos conectamos o servidor avísanos se estamos de acordo coa autenticación. Respostamos yes e pulsamos Enter.

root@kaliA:~# exit #Saír da consola remota ssh a que acabamos de acceder, para voltar á consola local de **root**.

root@kaliA:~# grep -n '^#X11DisplayOffset' /etc/ssh/sshd_config #Buscar no ficheiro /etc/ssh/sshd_config mediante o comando *grep* o patrón de texto '*#X11DisplayOffset*' nun comezo de liña ^ indicando o número/s da/s liña/s (*-n*) onde existe o patrón buscado.

root@kaliA:~# VAR=$(grep -n '^#X11DisplayOffset' /etc/ssh/sshd_config | cut -d ':' -f1 | tail -1) #Crear a variable VAR co contido o número da última liña (poden existir máis de 1 se o ficheiro xa fora modificado) do patrón buscado no ficheiro /etc/ssh/sshd_config

root@kaliA:~# sed -i "s/X11DisplayOffset/#X11DisplayOffset/g" /etc/ssh/sshd_config #Modificar no ficheiro /etc/ssh/sshd_config o patrón de texto *X11DisplayOffset* por *#X11DisplayOffset* en todas as concurrencias atopadas, inclusive que fosen na mesma liña, é dicir, comentar (mediante o caracter #) para que non teña lugar a directiva *X11DisplayOffset*

root@kaliA:~# sed -i "${VAR}a\X11DisplayOffset 100" /etc/ssh/sshd_config #Redirección gráfica no display 100 en conexións SSH, é dicir, modificar o ficheiro /etc/ssh/sshd_config engadindo a liña *X11DisplayOffset 100* logo da liña atopada anteriormente que comeza por *#X11DisplayOffset*

root@kaliA:~# /etc/init.d/ssh reload #Recargar o ficheiro de configuración do servidor SSH para que se activen os cambios realizados

root@kaliA:~# ssh -X kali@localhost #Realizar unha conexión SSH a localhost mediante o usuario kali e o seu contrasinal a través do porto TCP 22 solicitando a redirección gráfica (*-X*), agora no display 100. Se é a primeira ver que nos conectamos o servidor avísanos se estamos de acordo coa autenticación. Respostamos yes e pulsamos Enter.

kali@kaliA:~$ exit #Saír da consola remota ssh a que acabamos de acceder, para voltar á consola local de **root**.

root@kaliA:~# grep -n '^#X11UseLocalhost' /etc/ssh/sshd_config #Buscar no ficheiro /etc/ssh/sshd_config mediante o comando *grep* o patrón de texto '*#X11UseLocalhost*' nun comezo de liña ^ indicando o número/s da/s liña/s (*-n*) onde existe o patrón buscado.

root@kaliA:~# VAR=$(grep -n '^#X11UseLocalhost' /etc/ssh/sshd_config | cut -d ':' -f1 | tail -1) #Crear a variable VAR co contido o número da última liña (poden existir máis de 1 se o ficheiro xa fora modificado) do patrón buscado no ficheiro /etc/ssh/sshd_config

root@kaliA:~# sed -i "s/X11UseLocalhost/#X11UseLocalhost/g" /etc/ssh/sshd_config #Modificar no ficheiro /etc/ssh/sshd_config o patrón de texto *X11UseLocalhost* por *#X11UseLocalhost* en todas as concurrencias atopadas, inclusive que fosen na mesma liña, é dicir, comentar (mediante o caracter #) para que non teña lugar a directiva *X11UseLocalhost*

root@kaliA:~# sed -i "${VAR}a\X11UseLocalhost yes" /etc/ssh/sshd_config #Redirección gráfica onde soamente está permitida a interface loopback para exportación gráfica en conexións SSH, é dicir, modificar o ficheiro /etc/ssh/sshd_config engadindo a liña *X11UseLocalhost yes* logo da liña atopada anteriormente que comeza por *#X11UseLocalhost*

root@kaliA:~# /etc/init.d/ssh reload #Recargar o ficheiro de configuración do servidor SSH para que se activen os cambios realizados

root@kaliA:~# exit #Saír da consola local sudo na que estabamos a traballar para voltar á consola local de **kali**.

kali@kaliA:~$ ssh -X kali@localhost #Realizar unha conexión SSH a localhost mediante o usuario kali e o seu contrasinal a través do porto TCP 22 solicitando a redirección gráfica (*-X*), permitindo soamente a exportación gráfica a través de localhost. Se é a primeira ver que nos conectamos o servidor avísanos se estamos de acordo coa autenticación. Respostamos yes e pulsamos Enter.

kali@kaliA:~$ firefox & #Lanzar o navegador firefox, realizando a execución en segundo plano (&). Agora é posible mediante SSH visualizar comandos que empreguen o servidor gráfico debido a que temos activada a redirección gráfica.

kali@kaliA:~$ exit #Saír da consola remota ssh a que acabamos de acceder, para voltar á consola local de **root**.

Máquina virtual B: Kali amd64

NOTAS:

- **Cliente ssh GNU/Linux:**
  - Comando ssh. Paquete openssh-client (# apt update && apt -y install openssh-client).
  - Ficheiro de configuración: **/etc/ssh/ssh_config (man ssh_config)**

9. Pór contrasinal ao usuario, co que arranca a live, de nome: kali
   Na contorna gráfica abrir un terminal e executar:

   kali@kali:~$ setxkbmap es #Cambiar o mapa de teclado ao idioma español.
   kali@kali:~$ passwd kali || (echo -e 'kali\nkaliBpass\nkaliBpass' | passwd) #Cambiar o contrasinal do usuario kali. Por como contrasinal **kaliBpass**

10. Cambiar hostname da máquina virtual B. Por kaliB como hostname:
    kali@kali:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
    root@kali:~# echo 'kaliB' > /etc/hostname #Indicar ao sistema o valor do hostname.
    root@kali:~# echo 'kernel.hostname=kaliB' >> /etc/sysctl.conf #Indicar ao kernel o valor do hostname.
    root@kali:~# sysctl -p #Activar o cambio de hostname sen ter que pechar sesión nin reiniciar
    root@kali:~# exit #Saír da consola local sudo na que estabamos a traballar para voltar á consola local de **kali**.
    kali@kali:~$ exit #Pechar o terminal saíndo da consola local do usuario **kali**.

11. Configurar a rede:
    Na contorna gráfica abrir un terminal e executar:

    kali@kaliB:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
    root@kaliB:~# /etc/init.d/avahi-daemon stop #Parar o demo avahi-daemon(control resolución de nomes) para poder configurar de forma manual a configuración de rede e non ter conflicto con este demo.
    root@kaliB:~# /etc/init.d/network-manager stop #Parar o demo network-manager(xestor de rede) para poder configurar de forma manual a configuración de rede e non ter conflicto con este xestor.
    root@kaliB:~# ip addr show #Amosar a configuración de todas as tarxetas de rede. Nesta caso, na máquina A, as tarxetas de redes: loopback(lo) e interna(eth0).
    root@kaliB:~# ip addr add 192.168.120.101/24 dev eth0 #Configurar a tarxeta de rede interna eth0, coa IP: 192.168.120.101 e máscara de subrede: 255.255.255.0.
    root@kaliB:~# ip addr show #Amosar a configuración de todas as tarxetas de rede. Nesta caso, na máquina A, as tarxetas de redes: loopback(lo) e interna(eth0).
    root@kaliB:~# ping -c4 192.168.120.101 #Comprobar mediante o comando ping a conectividade coa interface de rede local eth0
    root@kaliB:~# ping -c4 192.168.120.100 #Comprobar mediante o comando ping a conectividade coa Máquina Virtual A na IP 192.168.120.100
    root@kaliB:~# exit #Saír da consola local sudo na que estabamos a traballar para voltar á consola local de **kali**.

12. **Comprobar estado do Servidor SSH e execución de comandos a través da conexión SSH:**
    kali@kaliB:~$ nc -vz 192.168.120.100 22 #Mediante o comando nc(netcat) comprobar se o porto 22 do servidor ssh (192.168.120.100) está en estado escoita(listen), esperando conexións. A opción -v corresponde á opción verbose, o que permite amosar información máis detallada na saída do comando. A opción -z permite devolver PROMPT do sistema e de igual xeito facer o escaneo ao/s porto/s solicitados. O número 22 é o porto TCP a escanear.
    kali@kaliB:~$ ssh kali@192.168.120.100 #Comprobar se o servidor SSH está activo e podemos conectarnos a el en 192.168.120.100 co usuario kali e o seu contrasinal no porto TCP 22. Se é a primeira ver que nos conectamos o servidor avísanos se estamos de acordo coa autenticación. Respostamos yes e pulsamos Enter.
      kali@kaliA:~$ exit #Saír da consola remota ssh a que acabamos de acceder, para voltar á consola local de **root**.
    kali@kaliB:~$ ssh kali@192.168.120.100 "netstat -natp | grep 22" #Executar o comando **netstat** no servidor SSH 192.168.120.100 e ver a saída da execución do comando na máquina local kaliB.
    kali@kaliB:~$ ssh kali@192.168.120.100 ss -natp #Executar o comando **ss** no servidor SSH 192.168.120.100 e ver a saída da execución do comando na máquina local kaliB.
    kali@kaliB:~$ ssh -X kali@192.168.120.100 "thunar &" #Lanzar en segundo plano o explorador de arquivos thunar da máquina 192.168.120.100 a través dunha conexión SSH e como está activada a redirección gráfica é posible visualizar na máquina local (kaliB) o administrador de arquivos da máquina virtual A (kaliA).

```
kali@kaliB:~$ ssh -X kali@192.168.120.100 "firefox &" #Lanzar en segundo plano o navegador firefox da máquina 192.168.120.100 a través dunha conexión SSH e como está activada a redirección gráfica é posible visualizar na máquina local (kaliB) o navegador gráfico da máquina virtual A (kaliA).
```

Ricardo Feijoo Costa