Cifrado asimétrico: Certificado Apache

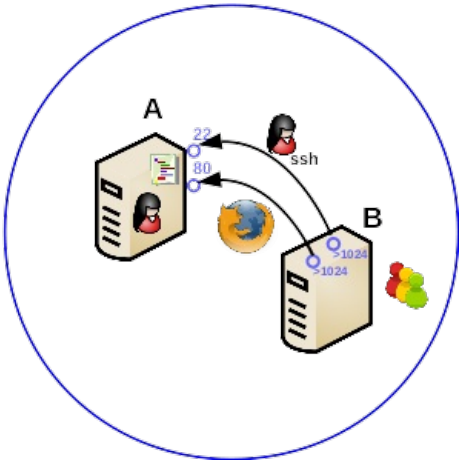
ESCENARIO

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

Máquina virtual A:
Rede Interna
Servidor SSH: openssh-server
Servidor Web: Apache (apache2)
ISO: Kali Live amd64
IP/MS: 192.168.120.100/24

Máquina virtual B:
Rede Interna
Cliente SSH: openssh-client (ssh)
Cliente Web: Navegador (firefox)
ISO: Kali Live amd64
IP/MS: 192.168.120.101/24

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



LIMITACIÓN DE RESPONSABILIDADE O autor do presente documento declina calquera responsabilidade asociada ao uso incorrecto e/ou malicioso que puidese realizarse coa información exposta no mesmo. Por tanto, non se fai responsable en ningún caso, nin pode ser considerado legalmente responsable en ningún caso, das consecuencias que poidan derivarse da información contida nel ou que esté enlazada dende ou hacia el, incluíndo os posibles erros e información incorrecta existentes, información difamatoria, así como das consecuencias que se poidan derivar sobre a súa aplicación en sistemas de información reais e/ou virtuais. Este documento foi xerado para uso didáctico e debe ser empregado en contornas privadas e virtuais controladas co permiso correspondente do administrador desas contornas.

NOTAS:

- Cliente ssh GNU/Linux: **comando ssh (paquete openssh-client)**
- Documentación oficial sobre o Servidor web **Apache (v2.4)**
- Referencia Apache: **Apache (v2.4)**
- Orde resolución DNS: **nsswitch**
- Referencia uso **Wireshark**

Práctica Cifrado asimétrico: Certificado Apache

Máquina virtual A: Kali amd64

1. Na contorna gráfica abrir un terminal e executar:
   kali@kali:~$ setxkbmap es #Cambiar o mapa de teclado ao idioma español.
   kali@kali:~$ passwd kali #Cambiar o contrasinal do usuario kali. Por como contrasinal **abc123.** (Ollo que o contrasinal ten un caracter punto final).

2. Cambiar hostname da máquina virtual A. Por kaliA como hostname:

   OPCIÓN A:
   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**.

   OPCIÓN B:
   kali@kali:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
   root@kali:~# hostnamectl hostname kaliA || hostnamectl set-hostname kaliA #Modificar o hostname do sistema a kaliA.
   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:~$ setxkbmap es #Cambiar o mapa de teclado ao idioma español.
   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 || systemctl stop avahi-daemon #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 || pkill NetworkManager #Parar o demo network-manager(xestor de rede) ou o script NetworkManager (executado sen ser demo) 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.
      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 sudo na que estabamos a traballar para voltar á consola local de **kali**.
   kali@kaliA:~$

Máquina virtual B: Kali amd64

5. Configuración da rede. Na contorna gráfica abrir un terminal e executar:
   kali@kali:~$ setxkbmap es #Cambiar o mapa de teclado ao idioma español.
   kali@kali:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
   root@kali:~# /etc/init.d/avahi-daemon stop || systemctl stop avahi-daemon #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@kali:~# /etc/init.d/network-manager stop || pkill NetworkManager #Parar o demo network-manager(xestor de rede) ou o script NetworkManager (executado sen ser demo) para poder configurar de forma manual a configuración de rede e non ter conflicto con este xestor.
   root@kali:~# ip addr show #Amosar a configuración de todas as tarxetas de rede. Nesta caso, na máquina B as tarxetas de redes: loopback(lo) e interna(eth0).
   root@kali:~# 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@kali:~# ip addr show #Amosar a configuración de todas as tarxetas de rede. Nesta caso, na máquina B as tarxetas de redes: loopback(lo) e interna(eth0).
   root@kali:~# ping -c4 192.168.120.101 #Comprobar mediante o comando ping a conectividade coa interface de rede local eth0
   root@kali:~# ping -c4 192.168.120.100 #Comprobar mediante o comando ping a conectividade coa interface de rede da máquina virtual A
   root@kali:~# echo '192.168.120.100 kaliA' >> /etc/hosts #Engadir no ficheiro /etc/hosts, é dicir, na táboa estática de búsqueda para nomes de host (DNS) o nome kaliA, para que atenda á IP 192.168.120.100
   root@kali:~# ping -c4 kaliA #Comprobar mediante o comando ping a conectividade coa interface de rede da máquina virtual A

6. Cambiar hostname da máquina virtual B. Por kaliB como hostname:

   OPCIÓN A:
   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**.

   OPCIÓN B:
   kali@kali:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
   root@kali:~# hostnamectl hostname kaliB || hostnamectl set-hostname kaliB #Modificar o hostname do sistema a kaliB.
   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**.

7. **B** ^{SSH} **➜ A** Acceder mediante SSH á máquina virtual A dende a máquina virtual B. A partir de agora executaremos sempre os comandos dende a máquina virtual B, a través da consola SSH:
   Na contorna gráfica abrir un terminal e executar:

   kali@kaliB:~$ setxkbmap es #Cambiar o mapa de teclado ao idioma español.
   kali@kaliB:~$ nc -vz 192.168.120.100 22 #Mediante o comando nc(netcat) comprobar que 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.
   kali@kaliB:~$ nc -vz kaliA 22 #Mediante o comando nc(netcat) comprobar que 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.
   kali@kaliB:~$ ssh -v kali@192.168.120.100 #Comprobar se o servidor SSH está activo e podemos conectarnos a el. Agora accedemos como o usuario **kali** a través da conexión cifrada SSH.
   kali@kaliA:~$

8. Activar Servidor Web Apache:

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/apache2 status #Comprobar o estado do servidor web Apache.
root@kaliA:~# /etc/init.d/apache2 start #Iniciar o servidor web Apache.
root@kaliA:~# /etc/init.d/apache2 status #Comprobar o estado do servidor web Apache.
root@kaliA:~# nc -vz 192.168.120.100 80 #Mediante o comando nc(netcat) comprobar se o porto 80 do servidor web Apache 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 80 é o porto TCP a escanear.

No caso da distribución Kali xa temos instalado o servidor web Apache, pero nunha distribución baseada en Debian poderiamos instalalo do seguinte xeito:

apt update #Actualizar o listado de paquetes dos repositorios (/etc/apt/sources.list, /etc/apt/sources.list.d/)
apt search apache2 #Buscar calquera paquete que coincida co patrón de búsqueda apache2
apt -y install apache2 #Instalar o paquete apache2, é dicir, instalar o servidor HTTP apache2. Co parámetro -y automaticamente asumimos yes a calquera pregunta que ocorra na instalación do paquete.

9. Lanzar na máquina virtual B (Kali) un navegador e visitar a IP 192.168.120.100 ou a URL http://192.168.120.100

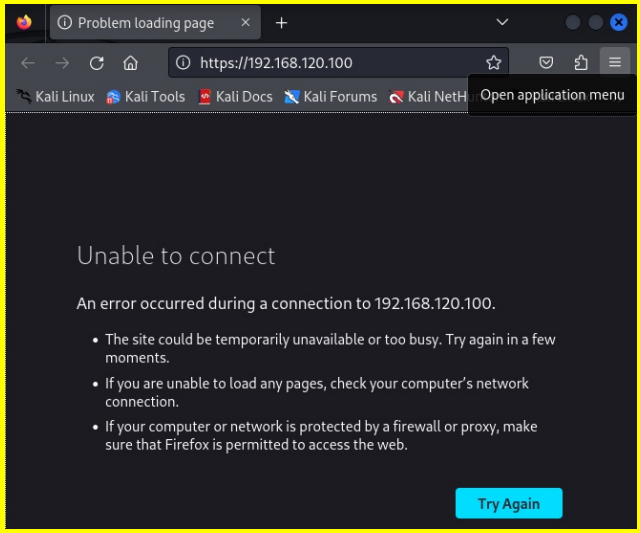
10. Permisos apache:

root@kaliA:~# chown -R www-data. /var/www/html/ #Cambiar usuario propietario www-data e grupo propietario www-data a toda a árbore de ficheiros e directorios que colgan do directorio DocumentRoot de Apache: /var/www/html
root@kaliA:~# chmod 444 /var/www/html/index.html #Cambiar a só lectura os permisos **ugo** do ficheiro index.html situado en /var/www/html, é dicir, establecer os permisos r--r--r-- (soamente lectura para o usuario propietario, o grupo propietario e o resto do mundo)
root@kaliA:~# /etc/init.d/apache2 restart #Reiniciar o servidor web Apache.
root@kaliA:~# /etc/init.d/apache2 status #Comprobar o estado do servidor web Apache.

11. Actualizar na máquina virtual B (Kali) a páxina referente á URL http://192.168.120.100

12. Lanzar no navegador da máquina virtual B (Kali) unha nova lapela coa URL **https://192.168.120.100** Que acontece? Por que?

Que non se pode visualizar a páxina obtendo un erro. Isto é debido a que o servidor web Apache non está configurado (aínda) para servir páxinas a través do protocolo HTTPS (porto TCP 443 por estándar)

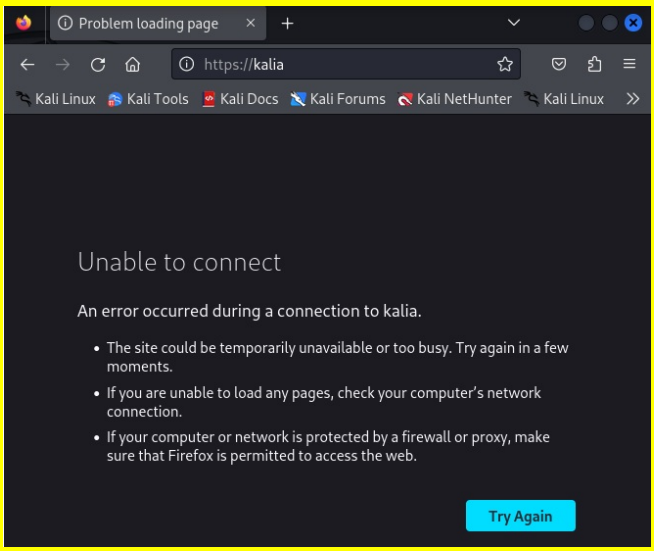


E se visitamos a URL **https://kaliA** Que acontece? Por que?

Pois o mesmo:
- Que non se pode visualizar a páxina obtendo un erro. Isto é debido a que o servidor web Apache non está configurado (aínda) para servir páxinas a través do protocolo HTTPS (porto TCP 443 por estándar)

Soamente que agora está resolvendo o nome kaliA á IP 192.168.120.100 a través do ficheiro /etc/hosts:

```
$ grep hosts /etc/nsswitch.conf
hosts:          files mdns4_minimal [NOTFOUND=return] dns
```



13. Activar configuración https (módulo SSL, porto TCP 443) en Apache:

Revisar o contido dos directorios:
- /etc/apache2/sites-available
- /etc/apache2/sites-enabled

root@kaliA:~# nc -vz 192.168.120.100 443 #Mediante o comando nc(netcat) comprobar se o porto 443 do servidor web Apache 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 443 é o porto TCP a escanear.

root@kaliA:~# a2ensite default-ssl #Habilitar o VirtualHost default-ssl, que configura o acceso a través de https (porto TCP 443)

root@kaliA:~# /etc/init.d/apache2 reload #Recargar a configuración do servidor web Apache.

Revisar o contido dos directorios:
- /etc/apache2/sites-available
- /etc/apache2/sites-enabled

root@kaliA:~# nc -vz 192.168.120.100 443 #Mediante o comando nc(netcat) comprobar se o porto 443 do servidor web Apache 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 443 é o porto TCP a escanear.

14. Lanzar de novo no navegador da máquina virtual B (Kali) unha nova lapela coa URL **https://192.168.120.100** Que acontece? Por que?

Pois o mesmo:
- Que non se pode visualizar a páxina obtendo un erro. Isto é debido a que o servidor web Apache non está completamente configurado (aínda) para servir páxinas a través do protocolo HTTPS (porto TCP 443 por estándar). Agora temos habilitado o sitio https no noso servidor web, pero non temos activado o Módulo SSL que permite a conexión a través do protocolo HTTPS.

E se visitamos a URL **https://kaliA** Que acontece? Por que?

Pois o mesmo, soamente que agora está resolvendo o nome kaliA á IP 192.168.120.100 a través do ficheiro /etc/hosts:

```
$ grep hosts /etc/nsswitch.conf
hosts:          files mdns4_minimal [NOTFOUND=return] dns
```

15. Activar certificado https (módulo SSL, porto TCP 443) en Apache:

root@kaliA:~# nc -vz 192.168.120.100 443 #Mediante o comando nc(netcat) comprobar se o porto 443 do servidor web Apache 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 443 é o porto TCP a escanear.

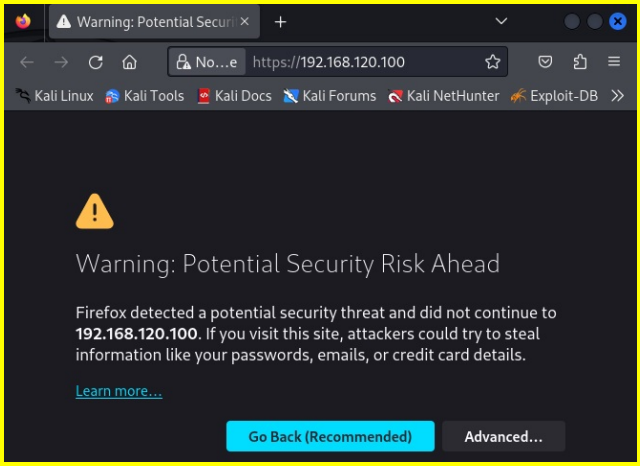
root@kaliA:~# a2enmod ssl #Habilitar o módulo ssl que permite activar a configuración do VirtualHost default-ssl, que configura o acceso a través de https (porto TCP 443)

root@kaliA:~# /etc/init.d/apache2 restart #Reiniciar a configuración do servidor web Apache.

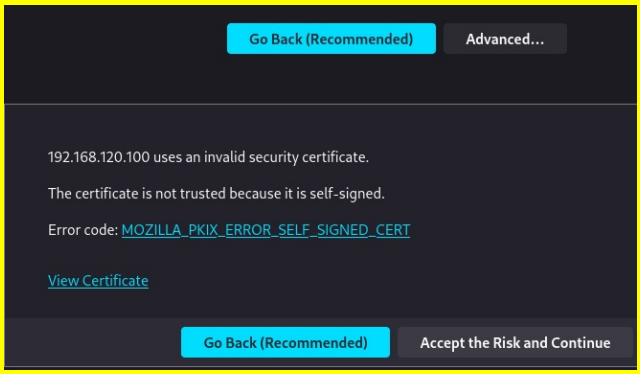
root@kaliA:~# nc -vz 192.168.120.100 443 #Mediante o comando nc(netcat) comprobar se o porto 443 do servidor web Apache está en estado escoita(listen), esperando conexións. A opción -v corresponde á opción verbose, o que permite amosar información máis detallada na saída do comando. A opción -z permite devolver PROMPT do sistema e de igual xeito facer o escaneo ao/s porto/s solicitados. O número 443 é o porto TCP a escanear.

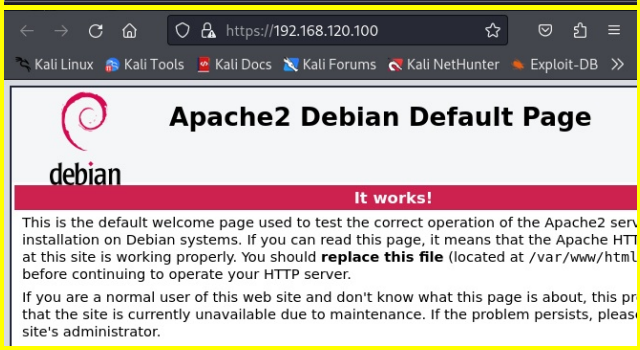
16. Lanzar de novo no navegador da máquina virtual B (Kali) unha nova lapela coa URL **https://192.168.120.100** Que acontece? Por que?

Pois agora pódese visualizar a páxina a través do protocol HTTPS porque o servidor Apache está configurao e posúe o módulo activado.



Agora debemos aceptar o certificado para poder ver o contido da páxina, posto que como este certificado está autoasinado por Apache e non por unha entidade certificadora(CA) o navegador revisa o contedor de certificados de CA que ten validados e como non se correponde con ningunha amosar una mensaxe de aviso **(Warning)**. Así, picamos en **Advanced** e logo en **Accept the Risk and Continue**

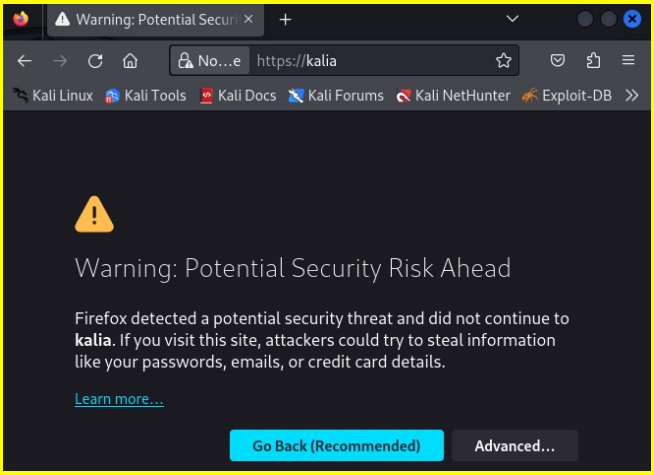


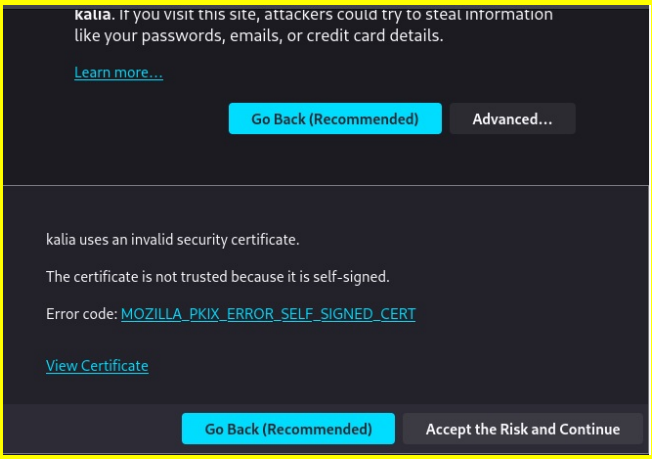


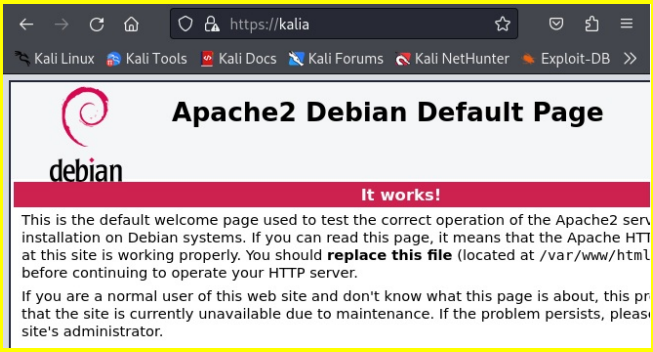
E se visitamos a URL **https://kaliA** Que acontece? Por que?

Pois o mesmo, agora pódese visualizar a páxina a través do protocol HTTPS porque o servidor Apache está configurao e posúe o módulo activado, soamente que agora está resolvendo o nome kaliA á IP 192.168.120.100 a través do ficheiro /etc/hosts:

```
$ grep hosts /etc/nsswitch.conf
hosts:          files mdns4_minimal [NOTFOUND=return] dns
```

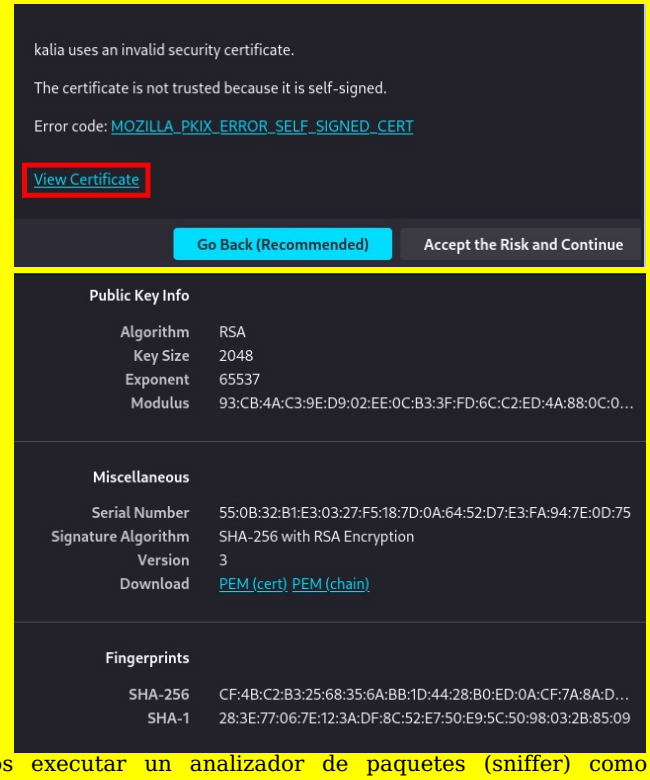




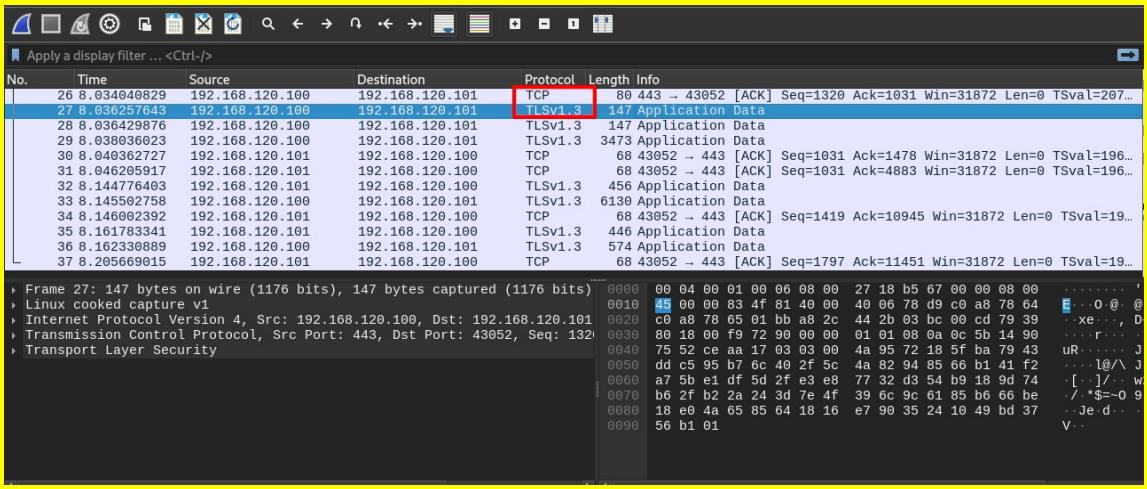


17. É a conexión segura? A transmisión da información realízase mediante cifrado (RSA, DSA...) (MD5, SHA1, SHA-256...)?

NON, picamos en **View Certificate** para visualizar as propiedades do certificado, nas cales podemos comproar os algoritmos de cifrado utilizado no certificado para garanir a autenticidade do servidor pero non na transmisión de información:



Para verificalo podemos executar un analizador de paquetes (sniffer) como Wireshark, atopando que a comunicación ten lugar mediante TCP e TLSv1.3:



Isto significa que cando estableces unha conexión TCP/TLSv1.3 cun servidor Apache 2.4, estás:

- Establecendo unha conexión confiable: TCP garantiza que os datos envíense e reciban correctamente.
- Protexendo os teus datos: TLSv1.3 cifra a comunicación, o que significa que os teus datos están protexidos de miradas indiscretas.
- Verificando a identidade do servidor: Asegúraste que estás conectado ao servidor web correcto e non a un impostor.
- Utilizando un protocolo seguro e eficiente: TLSv1.3 é a última versión do protocolo TLS e ofrece as mellores prácticas de seguridade.

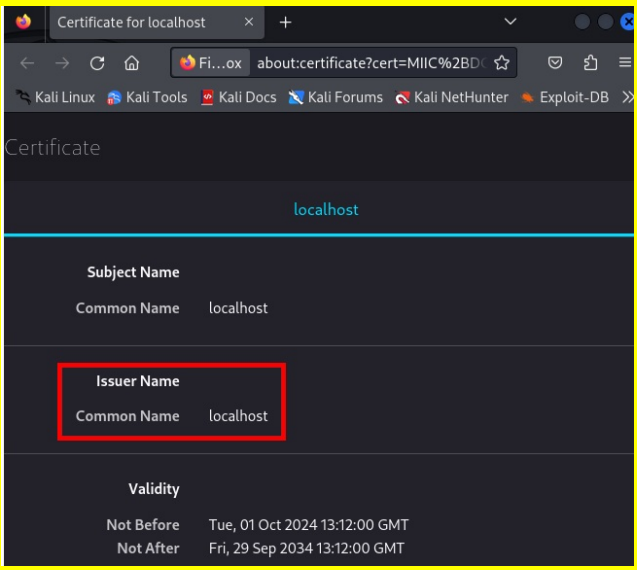
Así TLSv1.3, cando se implementa correctamente, proporciona un alto nivel de garantía para as propiedades **CAIN (Confidencialidade, Autenticidade, Integridade e Non Repudio)** nas comunicacións de rede. Desglose de cada propiedade:

- Confidencialidade: TLSv1.3 utiliza cifrados fortes para garantir que a información transmitida entre o cliente e o servidor sexa ilexible para calquera observador non autorizado. Isto protexe a privacidade dos datos.
- Autenticidade: Aínda que un certificado autofirmado limita a autenticación á confianza que o cliente deposite no servidor, TLSv1.3, en xeral, permite verificar a identidade do servidor a través de certificados dixitais emitidos por Autoridades de Certificación (CA) de confianza. Isto asegura que o cliente estase comunicando co servidor correcto e non cun impostor.
- Integridade: TLSv1.3 utiliza códigos de autenticación de mensaxes(MAC) para detectar calquera modificación non autorizada dos datos durante a transmisión. Se se altera un so bit, o MAC non coincidirá e a conexión interrumpirase.
- Non repudio: Aínda que TLSv1.3 non proporciona unha proba irrefutable de non repudio en todos os escenarios, o protocolo axuda a establecer un historial das comunicacions. Isto pode ser utilizado como evidencia en caso de disputas, xa que demostra que certas accións o transaccións tiveron lugar.

En poucas palabras: Ao utilizar TCP e TLSv1.3, estás navegando de forma segura e privada.

O navegador empregado confía no certificado configurado no servidor Apache? Ese certificado está asinado por unha entidade certificadora?

NON, de aí que aparecera a mensaxe anterior de **Warning**. De feito se revisamos as propiedades do certificado obtemos que o certificado é autofirmado por **localhost** e **localhost é calquera**



Podemos ver información sobre o certificado de Apache a través do navegador(Si/Non)? Se é posible como se pode revisar esa información?

SI, basicamente a través do propio candado que aparece antes da URL ou ben a través de **View Certificate** comentado anteriormente: