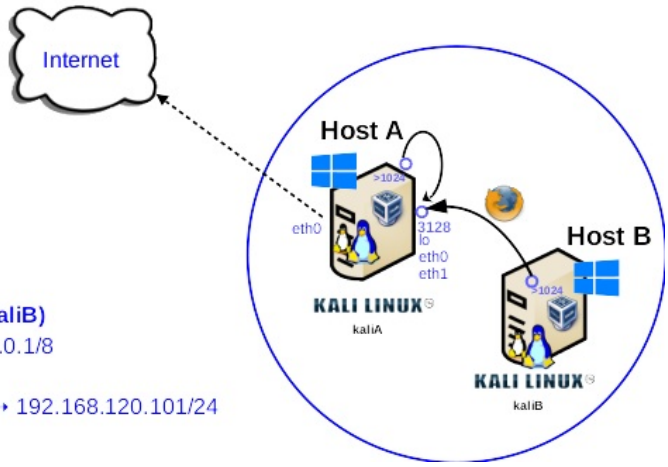
# Servizo Proxy Caché: Squid

**ESCENARIO**

**Máquinas virtuais:**
RAM ≤ 2048MB  CPU ≤ 2  PAE/NX habilitado
ISO: Kali Live amd64
BIOS: Permite arranque dispositivo extraíble: CD/DVD, USB
Cliente Web: Navegador (firefox)

**Máquina virtual A (kaliA)**
loopback (lo) → 127.0.0.1/8
Rede (eth0): NAT → 10.0.2.15/24
Rede (eth1): Interna → 192.168.120.100/24
Servidor Proxy Caché Squid → Porto TCP 3128

**Máquina virtual B (kaliB)**
loopback (lo) → 127.0.0.1/8
Rede (eth0): Interna → 192.168.120.101/24




**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**:

- Documentación oficial sobre o Servidor Proxy Caché **Squid**
  - Paquete squid (# apt update && apt -y install squid).
  - Configuración en: **/etc/squid/ (man squid)**

    **squid.conf ➜** Ficheiro de configuración principal. Non debería modificarse con novas directivas. Así, se se quere extender a configuración global de Squid deberianse incluír noutros ficheiros de configuración coa extensión **.conf** dentro do directorio **/etc/squid/conf.d**

    Aparecen por liña valores: directiva e argumentos. As directivas poder consultarse en **Reference**.

    Directivas de vital importancia son:

    **acl ➜** Definir listas de control de acceso (ACLs). ACLs: all, manager, localhost, to_localhost, to_linklocal e CONNECT están predefinidas.

    **http_access ➜** Permitir/Denegar lista de control de acceso (ACLs). NOTA sobre os valores predeterminados: Se non hai liñas "allow", o predeterminado é denegar a solicitude. Se ningunha das liñas de "allow" provoca unha coincidencia, o predeterminado é o oposto á última liña da lista. Se a última liña fose "deny", o valor predeterminado é "allow". Pola contra, se a última liña é "allow", o valor predeterminado será "deny". Por estes motivos, trátase dunha boa idea ter unha entrada "deny all" ao final das liñas http_access para evitar posibles confusións.

    **include ➜** Engadir ficheiros de configuración.

    **http_port ➜** Onde se configura o hostname/IP e o porto TCP de acceso. Por defecto soamente aparece o porto: **3128**, o cal indica que calquera hostname/IP que posúa o sistema estará a escoita nese porto.

    **coredump_dir ➜** Especificar o directorio para gardar os volcados de memoria.
    Por defecto: **/var/spool/squid/**

    **refresh_pattern ➜** Definir expresión regular(regex).

  - **Squid Log Files: /var/log/squid/**

Configuración por defecto **squid.conf**:

kali@kali:~$ **grep -v '^#' /etc/squid/squid.conf | sed '/^$/d'**

```
acl localnet src 0.0.0.1-0.255.255.255  # RFC 1122 "this" network (LAN)
acl localnet src 10.0.0.0/8             # RFC 1918 local private network (LAN)
acl localnet src 100.64.0.0/10          # RFC 6598 shared address space (CGN)
acl localnet src 169.254.0.0/16         # RFC 3927 link-local (directly plugged) machines
acl localnet src 172.16.0.0/12          # RFC 1918 local private network (LAN)
acl localnet src 192.168.0.0/16         # RFC 1918 local private network (LAN)
acl localnet src fc00::/7               # RFC 4193 local private network range
acl localnet src fe80::/10              # RFC 4291 link-local (directly plugged) machines
acl SSL_ports port 443
acl Safe_ports port 80          # http
acl Safe_ports port 21          # ftp
acl Safe_ports port 443         # https
acl Safe_ports port 70          # gopher
acl Safe_ports port 210         # wais
acl Safe_ports port 1025-65535  # unregistered ports
acl Safe_ports port 280         # http-mgmt
acl Safe_ports port 488         # gss-http
acl Safe_ports port 591         # filemaker
acl Safe_ports port 777         # multiling http
http_access deny !Safe_ports
http_access deny CONNECT !SSL_ports
http_access allow localhost manager
http_access deny manager
include /etc/squid/conf.d/*.conf
http_access allow localhost
http_access deny all
http_port 3128
coredump_dir /var/spool/squid
refresh_pattern ^ftp:           1440    20%     10080
refresh_pattern ^gopher:        1440    0%      1440
refresh_pattern -i (/cgi-bin/|\?) 0     0%      0
refresh_pattern .               0       20%     4320
```

kali@kali:~$ **cat /etc/squid/conf.d/debian.conf**

```
#
# Squid configuration settings for Debian
#

# Logs are managed by logrotate on Debian
logfile_rotate 0

# For extra security Debian packages only allow
# localhost to use the proxy on new installs
#
#http_access allow localnet
```

- URL de interese: **Asegurando Squid**

## Máquina virtual A: Kali amd64
## (lo -> loopback -> 127.0.0.1/8)
## (eth0 -> NAT -> 10.0.2.15/24)
## (eth1 -> Rede Interna -> 192.168.120.100/24)

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:
   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:~$ 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 #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 doutro xeito (co comando ip(ifconfig) de forma manual ou mediante networking (ficheiros /etc/init.d/networking, /etc/init.d/networking.d) 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), NAT(eth0) e interna(eth1).
   root@kaliA:~# ip addr add 192.168.120.100/24 dev eth1 #Configurar a tarxeta de rede interna eth1, 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), NAT(eth0) e interna(eth1).
   root@kaliA:~# ip route #Amosar a táboa de rutas do sistema.
   root@kaliA:~# cat /etc/resolv.conf #Ver o contido do ficheiro /etc/resolv.conf, o cal contén a configuración os servidores DNS a empregar para a resolución de nomes.
   root@kaliA:~# ping -c4 192.168.120.100 #Comprobar mediante o comando ping a conectividade coa interface de rede local eth1
   root@kaliA:~# ping -c4 www.google.es #Comprobar mediante o comando ping a conectividade co dominio www.google.es
   root@kaliA:~# 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@kaliA:~# ping -c4 kaliA #Comprobar mediante o comando ping a conectividade coa interface de rede da máquina virtual A

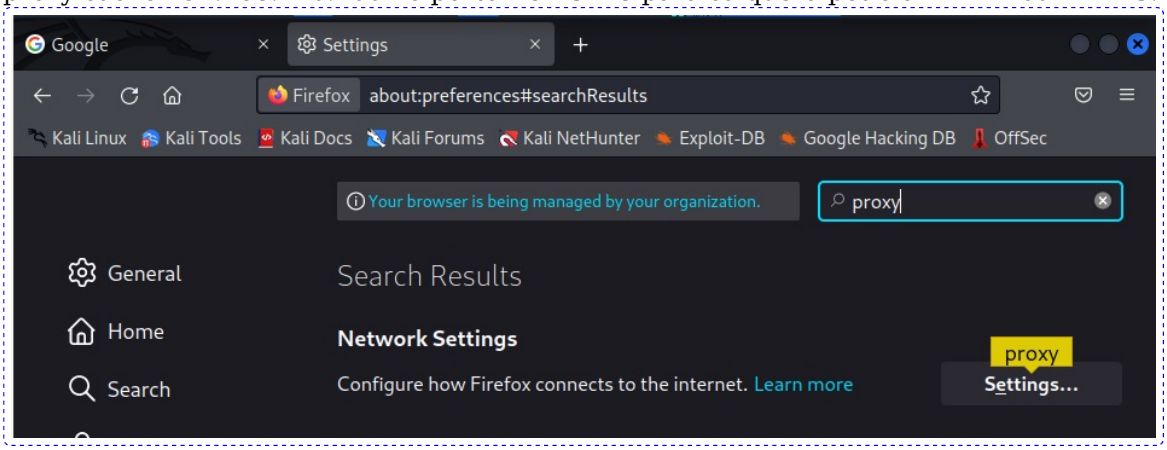
4. Activar Servidor Proxy Caché Squid:
   root@kaliA:~# apt update #Actualizar o listado de paquetes dos repositorios (/etc/apt/sources.list, /etc/apt/sources.list.d/)
   root@kaliA:~# apt search squid #Buscar calquera paquete que coincida co patrón de búsqueda squid
   root@kaliA:~# apt -y install squid #Instalar o paquete squid, é dicir, instalar o servidor proxy caché squid. Co parámetro -y automaticamente asumimos yes a calquera pregunta que ocorra na instalación do paquete.
   root@kaliA:~# /etc/init.d/squid status #Comprobar o estado do servidor proxy caché Squid.
   root@kaliA:~# /etc/init.d/squid start #Iniciar o servidor proxy caché Squid.
   root@kaliA:~# /etc/init.d/squid status #Comprobar o estado do servidor proxy caché Squid.
   root@kaliA:~# nc -vz 192.168.120.100 3128 #Mediante o comando nc(netcat) comprobar se o porto 3128 do servidor proxy caché Squid 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 3128 é o porto TCP a escanear.

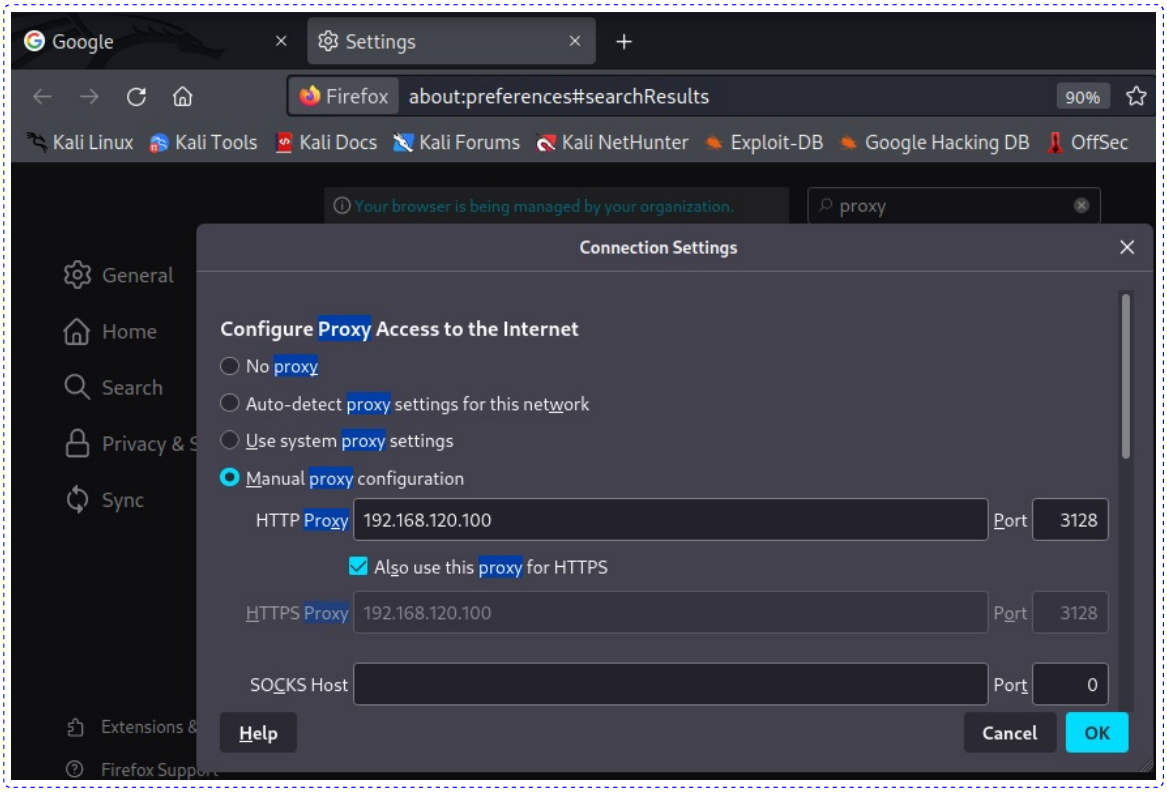
5. Lanzar na máquina virtual A (kaliA) un navegador e visitar a URL http://www.google.es

   Que acontece? Por que?

   Pois que é posible navegar e ver a URL http://www.google.es no navegador xa que a configuración de rede permite saír a Internet e resolver o dominio www.google.es, e a petición HTTP é redireccionada a HTTPS. Aínda non temos configurado no navegador a saída a través dun proxy.

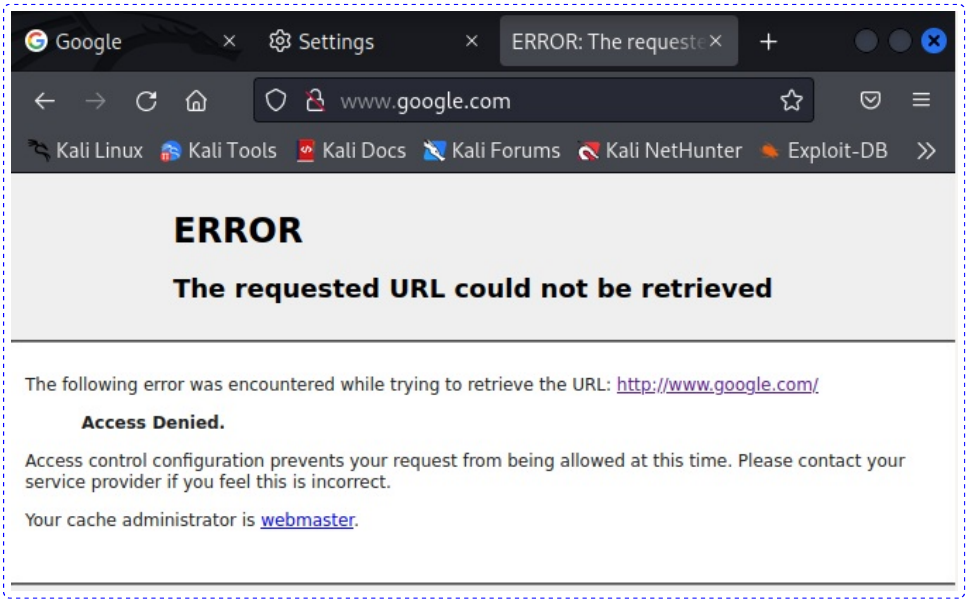
6. Na máquina virtual A (kaliA) configurar o navegador para que o acceso a Internet sexa a través do servidor proxy caché 192.168.120.100 no porto TCP 3128 para calquera petición HTTP ou HTTPS:




7. Na máquina virtual A (kaliA) abrir unha nova lapela na URL http://www.google.com

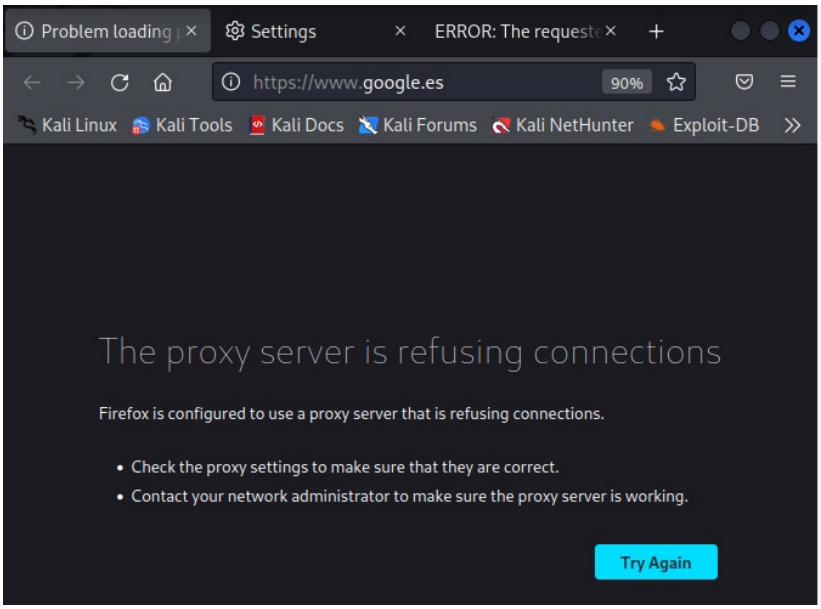
   Que acontece? Por que?




Agora temos configurado un proxy no navegador: 192.168.120.100(eth1), co cal a petición de saída faise á NIC eth1, e esta non ten permitido o acceso no proxy, é dicir, non existe ningunha ACL na configuración do proxy caché Squid que permita o acceso á NIC eth1 coa IP 192.168.120.100 ou ben a rede 192.168.120.0/24

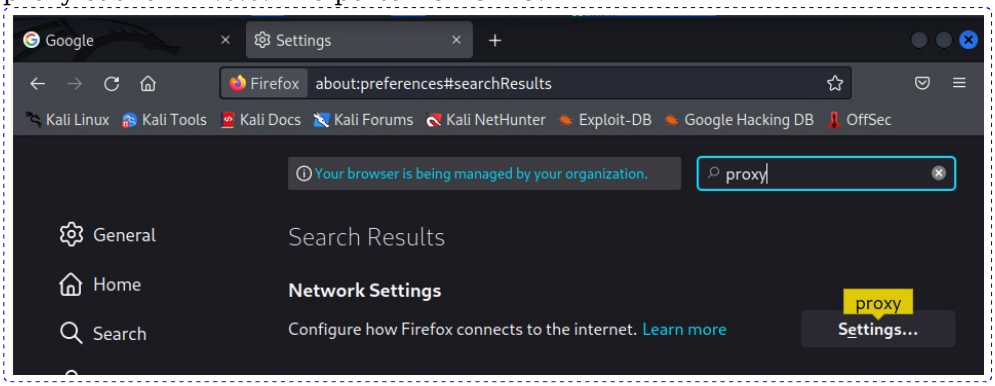
8. Actualizar na máquina virtual A (kaliA) a lapela referente á URL http://www.google.es

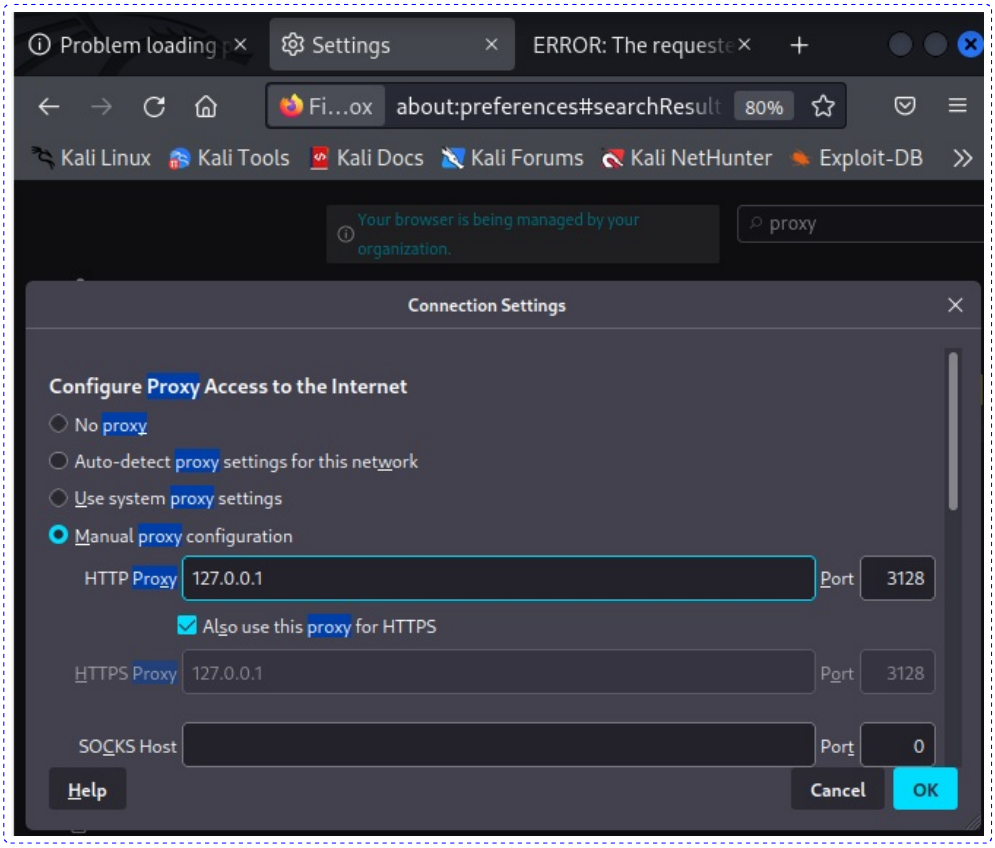
   Que acontece? Por que?




Pois se nos fixamos a petición anterior http://www.google.es redireccionou a https://www.google.es, e agora o proxy está a rexeitar esta petición HTTPS polo mesmo que comentamos antes: non existe ningunha ACL na configuración do proxy caché Squid que permita o acceso á NIC eth1 coa IP 192.168.120.100 ou ben a rede 192.168.120.0/24.

9. Na máquina virtual A (kaliA) configurar o navegador para que o acceso a Internet sexa a través do servidor proxy caché 127.0.0.1 no porto TCP 3128:



10. Na máquina virtual A (kaliA) abrir unha nova lapela na URL http://www.google.com

    Que acontece? Por que?

    Pois se nos fixamos a petición http://www.google.com redireccionou a https://www.google.com, e agora temos configurado un proxy no navegador: 127.0.0.1(lo), co cal a petición de saída faise á NIC loopback, e esta ten permitido o acceso no proxy, é dicir, existe algunha ACL na configuración do proxy caché Squid que permite o acceso a localhost (NIC lo):

    ```
    $ grep 127.0.0.1 /etc/hosts
    127.0.0.1       localhost kali
    $ grep localhost /etc/squid/squid.conf | grep -v '^#'
    http_access allow localhost manager
    http_access allow localhost
    ```

    Ademais é posible navegar e ver a URL https://www.google.com no navegador xa que agora estamos a solicitar na NIC loopback o proxy, co cal, está pode comunicarse coa NIC eth0, e polo tanto enrutar a Internet resolvendo o dominio www.google.com.

11. Actualizar na máquina virtual A (kaliA) a primeira lapela referente á URL http://www.google.es

    Que acontece? Por que?

    Pois, igual que no paso anterior se nos fixamos a petición http://www.google.es redireccionouse a https://www.google.es, e agora temos configurado un proxy no navegador: 127.0.0.1(lo), co cal a petición de saída faise á NIC loopback, e esta ten permitido o acceso no proxy, é dicir, existe algunha ACL na configuración do proxy caché Squid que permite o acceso a localhost (NIC lo):

    ```
    $ grep 127.0.0.1 /etc/hosts
    127.0.0.1       localhost kali
    $ grep localhost /etc/squid/squid.conf | grep -v '^#'
    http_access allow localhost manager
    http_access allow localhost
    ```

    Ademais é posible navegar e ver a URL https://www.google.es no navegador xa que agora estamos a solicitar na NIC loopback o proxy, co cal, está pode comunicarse coa NIC eth0, e polo tanto enrutar a Internet resolvendo o dominio www.google.es.

12. Reconfigurar Squid para permitir o acceso dende 192.168.120.100(rede Interna)(eth1):

    root@kaliA:~# sed -i 's/#http_access/http_access/' /etc/squid/conf.d/debian.conf #Descomentar a directiva **http_access allow localnet** no ficheiro de configuración /etc/squid/conf.d/debian.conf

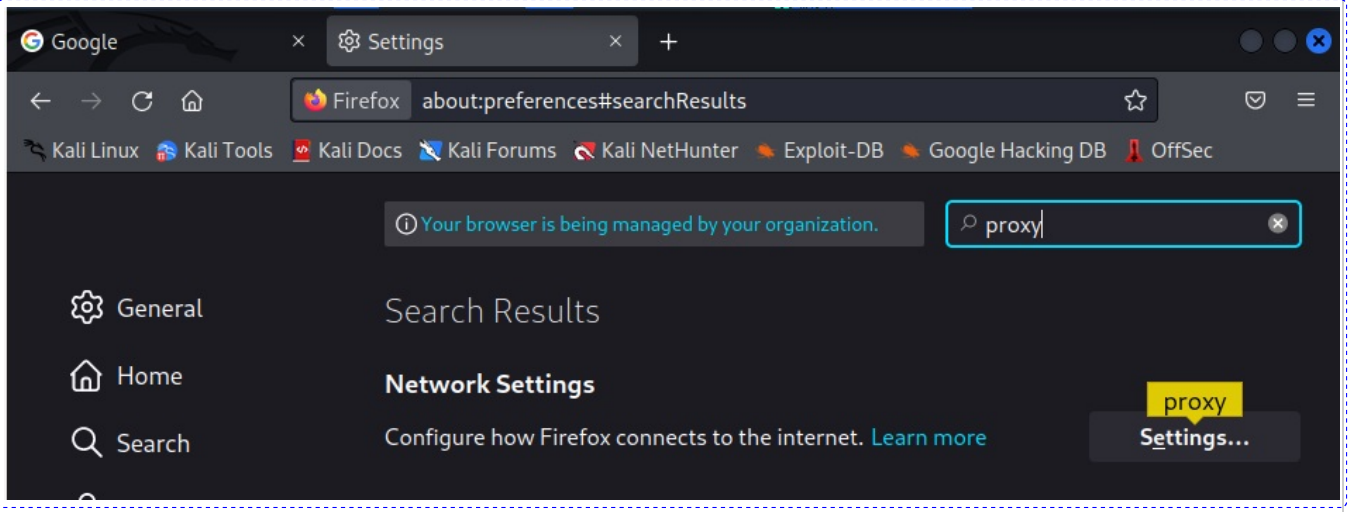
    Coa directiva anterior estamos a permitir todas as **acl localnet** seguintes:

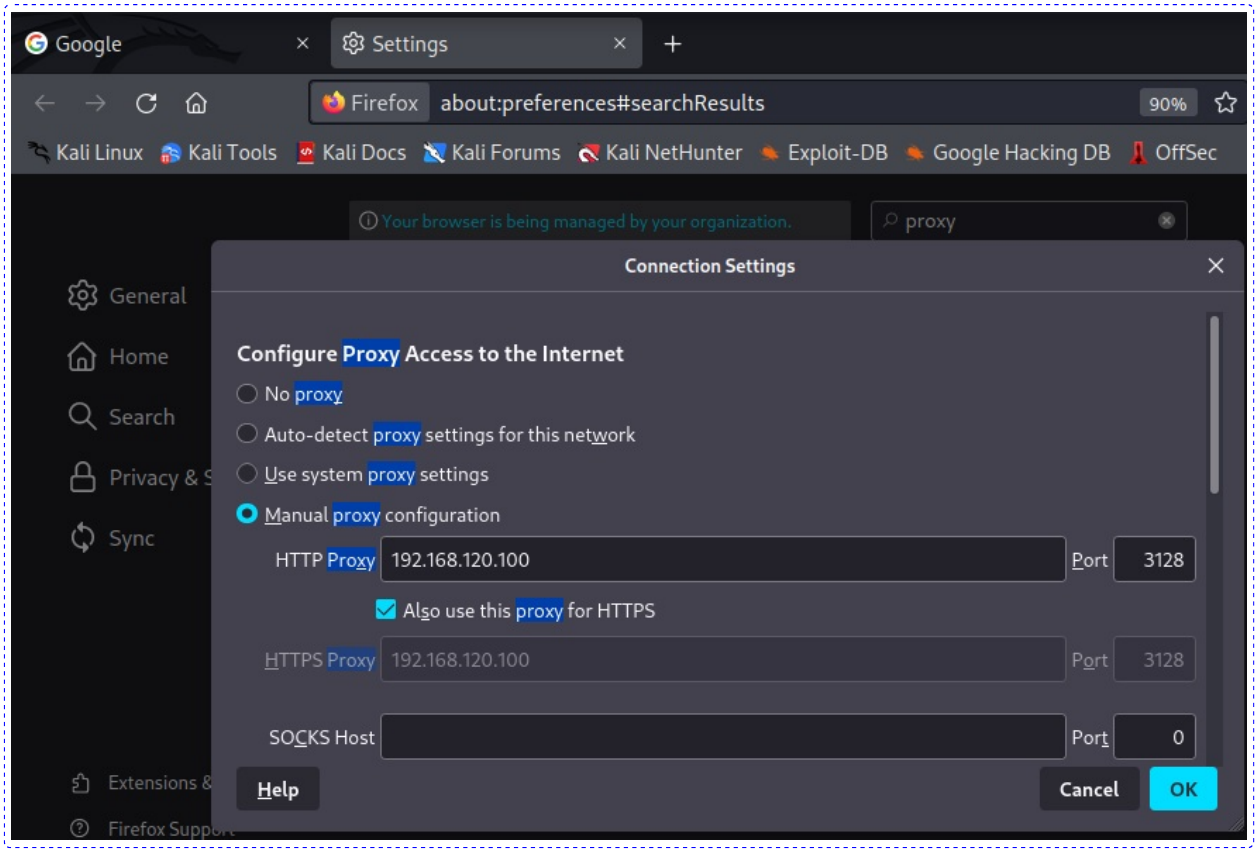
    ```
    $ grep 'acl localnet' /etc/squid/squid.conf
    acl localnet src 0.0.0.1-0.255.255.255  # RFC 1122 "this" network (LAN)
    acl localnet src 10.0.0.0/8             # RFC 1918 local private network (LAN)
    acl localnet src 100.64.0.0/10          # RFC 6598 shared address space (CGN)
    acl localnet src 169.254.0.0/16         # RFC 3927 link-local (directly plugged) machines
    acl localnet src 172.16.0.0/12          # RFC 1918 local private network (LAN)
    acl localnet src 192.168.0.0/16         # RFC 1918 local private network (LAN)
    acl localnet src fc00::/7               # RFC 4193 local private network range
    acl localnet src fe80::/10              # RFC 4291 link-local (directly plugged) machines
    ```

    É dicir, estamos a permitir a todas esas redes, e polo tanto, estamos a permitir á IP 192.168.120.100

    root@kaliA:~# /etc/init.d/squid reload #Recargar a configuración do Servidor Web Squid

    Reconfigurar o proxy no navegador:




13. Na máquina virtual A (kaliA) abrir unha nova lapela na URL http://www.edu.xunta.gal

    Que acontece? Por que?

    Pois, se nos fixamos a petición http://www.edu.xunta.gal redireccionouse a http://www.edu.xunta.gal/portal/, e agora temos configurado un proxy no navegador: 192.168.120.100(eth1), co cal a petición de saída faise á NIC eth1, e esta ten permitido o acceso no proxy, é dicir, existe algunha ACL na configuración do proxy caché Squid que permite o acceso á NIC eth1:

    ```
    $ grep 'acl localnet' /etc/squid/squid.conf | head -1
    acl localnet src 0.0.0.1-0.255.255.255  # RFC 1122 "this" network (LAN)
    ```

    Ademais é posible navegar e ver a URL http://www.edu.xunta.gal/portal/ no navegador xa que agora estamos a solicitar na NIC eth1 o proxy, e según a táboa de rutas a través da NIC eth0 o sistema pode enrutar a Internet resolvendo o dominio www.edu.xunta.gal

14. Actualizar na máquina virtual A (kaliA) a lapela referente á URL http://www.google.es

    Que acontece? Por que?

    Pois, como comentamos anteriormente agora é posible a sáida a Internet a través do proxy → eth1 → eth0 → Internet.

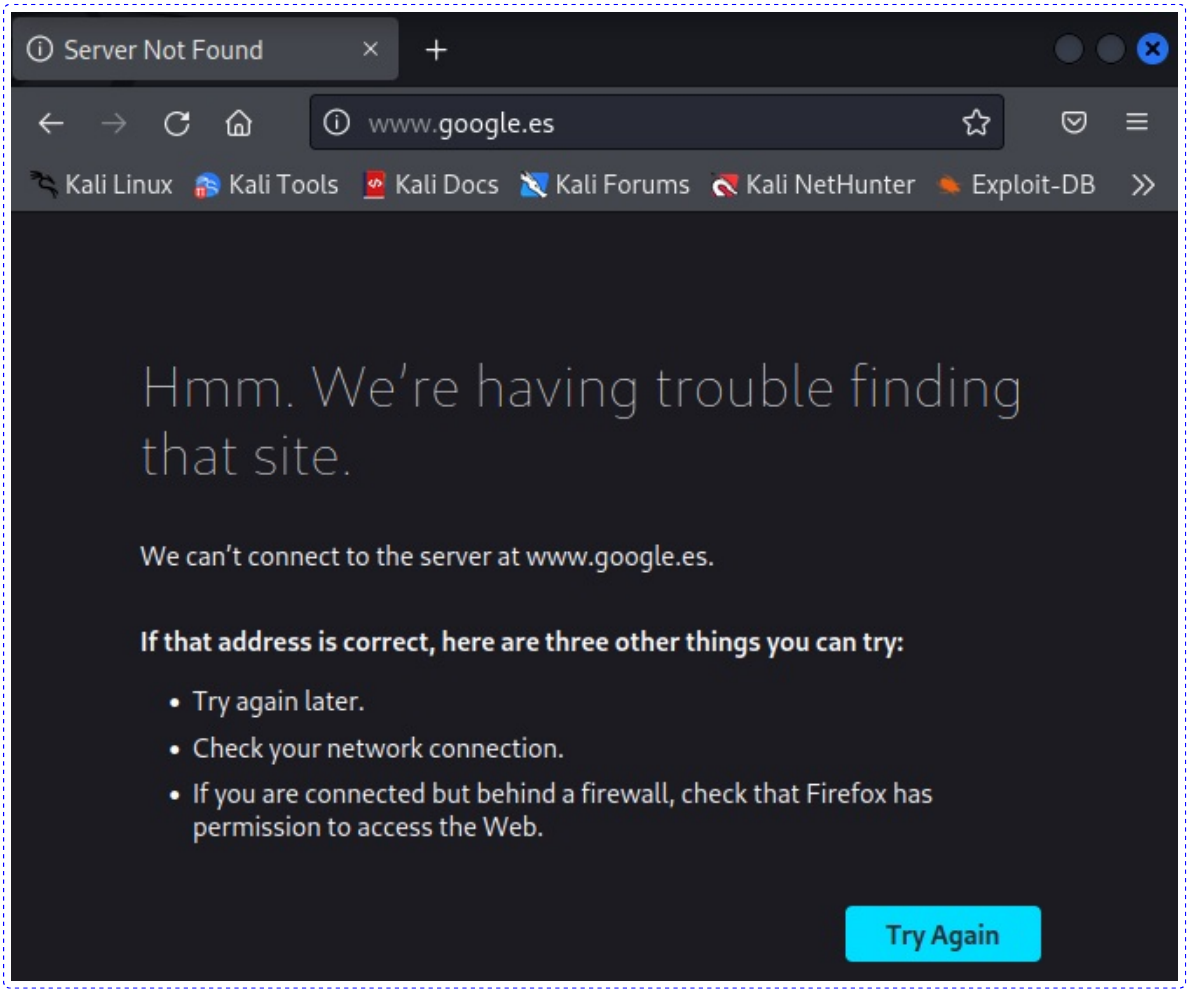
**Máquina virtual B: Kali amd64**
**(lo -> loopback -> 127.0.0.1/8)**
**(eth0 -> Rede Interna -> 192.168.120.101/24)**

15. 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 #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 doutro xeito (co comando ip(ifconfig) de forma manual ou mediante networking (ficheiros /etc/init.d/networking, /etc/init.d/networking.d) 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

16. Cambiar hostname da máquina virtual B. Por kaliB como hostname:
    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**.

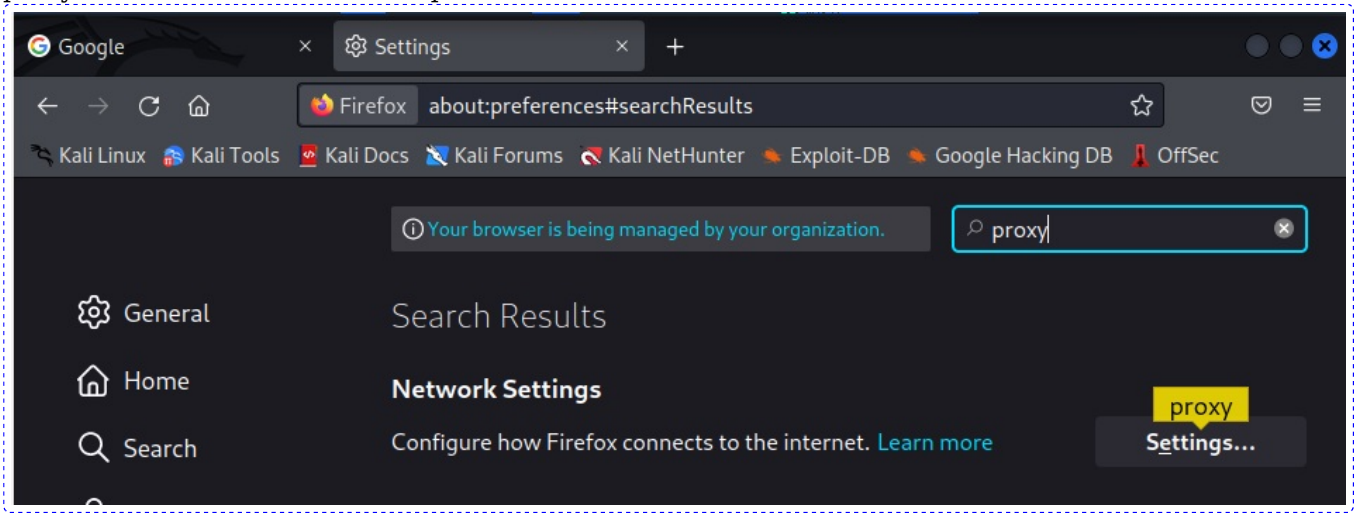
17. Lanzar na máquina virtual B (kaliB) un navegador e visitar a URL http://www.google.es

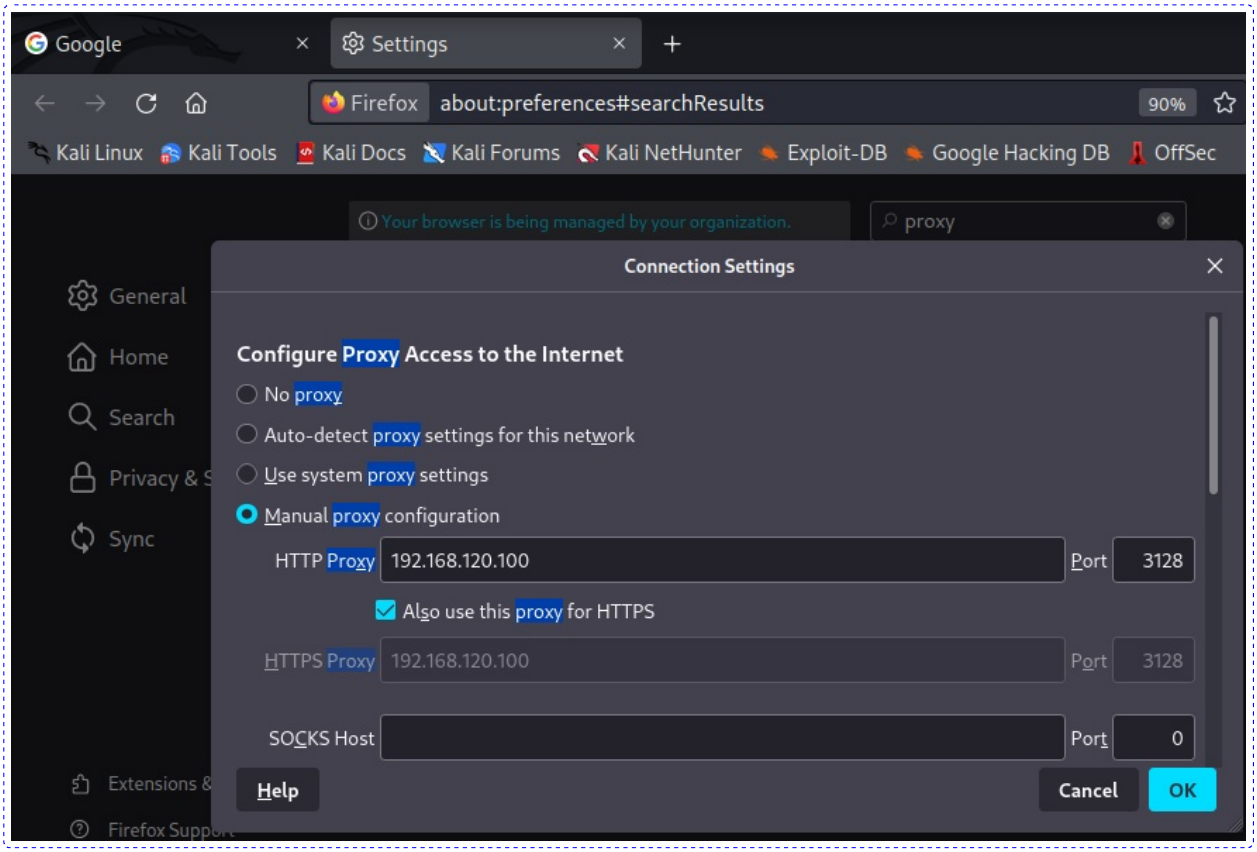
    Que acontece? Por que?



Pois que non é posible navegar e ver a URL http://www.google.es no navegador xa que a NIC eth0 está configurada en VirtualBox como rede interna, polo cal non ten acceso a Internet.

18. Na máquina virtual B (kaliB) configurar o navegador para que o acceso a Internet sexa a través do servidor proxy caché 192.168.120.100 no porto TCP 3128:




19. Actualizar na máquina virtual B (kaliB) a lapela referente á URL http://www.google.es

    Que acontece? Por que?

Pois que é posible navegar e ver a URL http://www.google.es no navegador xa que aínda que en kaliB a NIC eth0 está configurada en VirtualBox como rede interna, polo cal non ten acceso a Internet, a solicitude faise ao proxy configurado na NIC eth1 de kaliA, e xa vimos antes que esta NIC xa permitía o acceso a Internet. Así, a petición http://www.google.es redirecciónase a https://www.google.es, e agora temos configurado un proxy no navegador: 192.168.120.100(eth1 de kaliA), co cal a petición de saída faise á NIC eth1 de kaliA, e esta ten permitido o acceso no proxy, é dicir, existe algunha ACL na configuración do proxy caché Squid que permite o acceso á NIC eth1 de kaliA:

```
$ grep 'acl localnet' /etc/squid/squid.conf | head -1
acl localnet src 0.0.0.1-0.255.255.255  # RFC 1122 "this" network (LAN)
```

Ademais é posible navegar e ver a URL https://www.google.es no navegador xa que agora estamos a solicitar na NIC eth1 de kaliA o proxy, e según a táboa de rutas a través da NIC eth0 de kaliA o sistema pode enrutar a Internet resolvendo o dominio www.google.es

**Máquina virtual A: Kali amd64**
**(lo -> loopback -> 127.0.0.1/8)**
**(eth0 -> NAT -> 10.0.2.15/24)**
**(eth1 -> Rede Interna -> 192.168.120.100/24)**

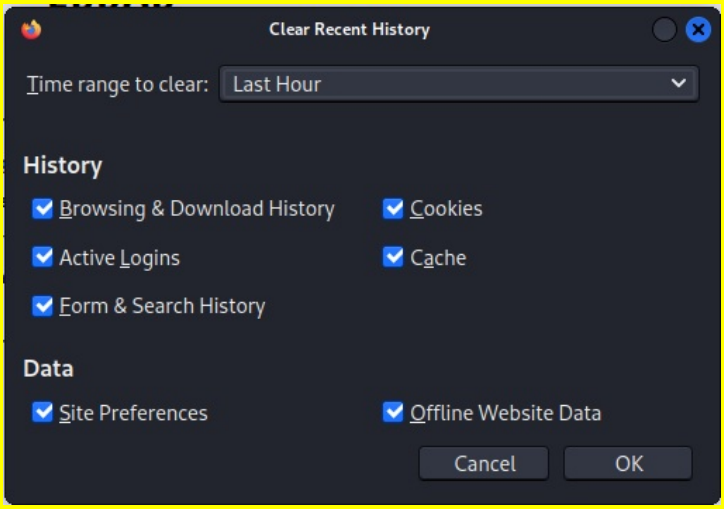
20. Na contorna gráfica abrir un terminal e executar:
    kali@kali:~$ setxkbmap es #Cambiar o mapa de teclado ao idioma español.
    kali@kali:~$ sudo su - #Acceder á consola de root(administrador) a través dos permisos configurados co comando sudo (/etc/sudoers, visudo)
        root@kaliA:~# sed -i 's/http_access allow/http_access deny/' /etc/squid/conf.d/debian.conf #Modificar no ficheiro de configuración /etc/squid/conf.d/debian.conf a directiva http_access allow localnet por http_access deny localnet

**Máquina virtual B: Kali amd64**
**(lo -> loopback -> 127.0.0.1/8)**
**(eth0 -> Rede Interna -> 192.168.120.101/24)**

21. Actualizar na máquina virtual B (kaliB) a lapela referente á URL http://www.google.es

    Que acontece? Por que?

    Pois se nos fixamos a petición anterior http://www.google.es redireccionou a https://www.google.es, e agora o proxy está a rexeitar tanto as peticións HTTP como HTTPS para a acl de nome localnet, pero segue sendo posible seguir vendo a URL solicitada no navegador xa que o navegador posúe a súa propia caché e segue amosando a páxina anterior sen realizar saída a Internet.
    Se limparamos a caché premendo ao mesmo tempo as teclas: <Shift>, <Ctrl> e <Supr>:

    E voltaramos a actualizar a lapela a URL http://www.google.es non se resolve.

    Abrir unha nova lapela na URL http://github.com

    Que acontece? Por que?

**Máquina virtual A: Kali amd64**
**(lo -> loopback -> 127.0.0.1/8)**
**(eth0 -> NAT -> 10.0.2.15/24)**
**(eth1 -> Rede Interna -> 192.168.120.100/24)**

22. 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@kaliA:~# /etc/init.d/squid reload #Recargar a configuración do Servidor Web Squid

**Máquina virtual B: Kali amd64**
**(lo -> loopback -> 127.0.0.1/8)**
**(eth0 -> Rede Interna -> 192.168.120.101/24)**

23. Actualizar de novo na máquina virtual B (kaliB) a lapela referente á URL http://www.google.es

    Que acontece? Por que?

Ricardo Feijoo Costa