

Análisis Forense 2

Wireshark

CC5325 - Taller de Hacking Competitivo

Protocolo de Internet

- **Capa de Enlace**
 - Wi-Fi, ARP, Ethernet, OSCP...
- **Capa de Internet**
 - IPv4, IPv6, ICMP
- **Capa de Transporte**
 - TCP, UDP
- **Capa de Aplicación**
 - HTTP, TLS, FTP, IMAP

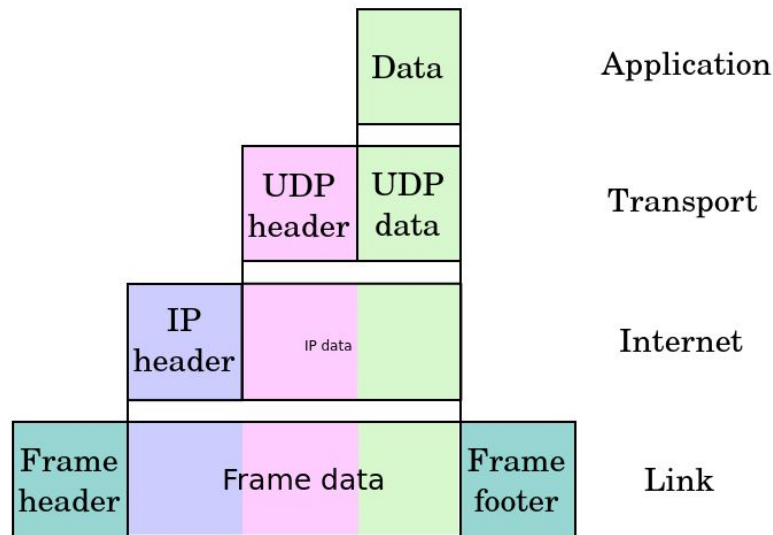
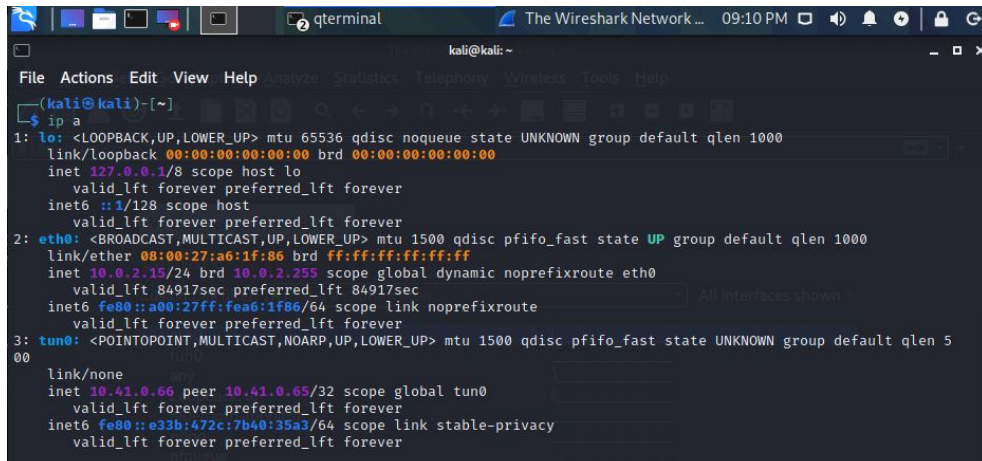


Imagen de Wikimedia Commons
creada por usuario **Cburnett**

Interfaces de Red (ip a)

- **eth0**: Interfaz "principal" de red
- **lo**: Interfaz virtual al mismo computador
- **tun0**: Interfaz virtual que representa la conexión por VPN
- **any**: Unión de todas las interfaces (en Wireshark)



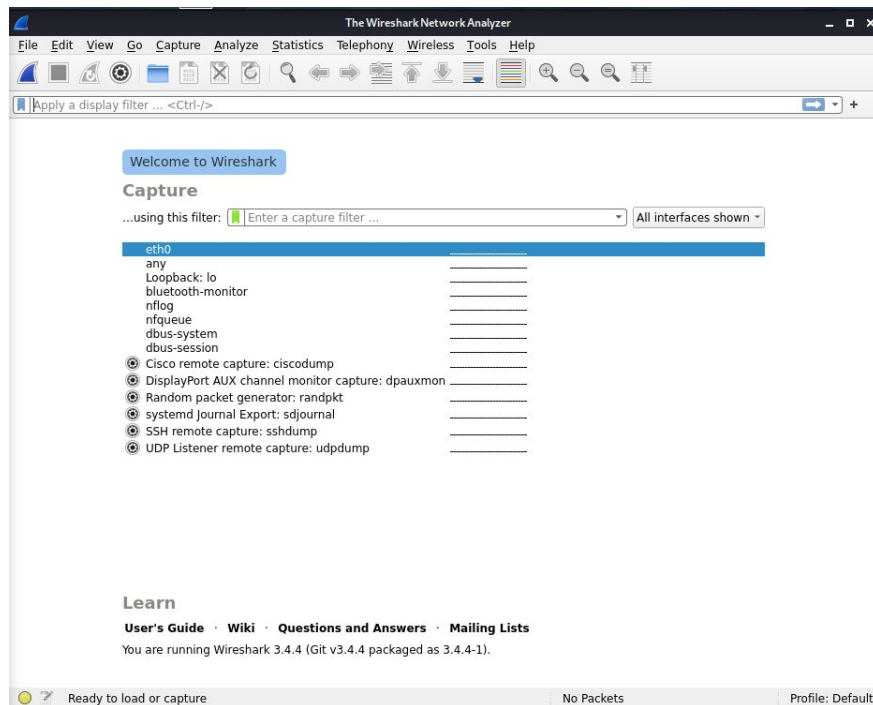
```
kali@kali: ~  
$ ip a  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
        valid_lft forever preferred_lft forever  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000  
    link/ether 08:00:27:a6:1f:86 brd ff:ff:ff:ff:ff:ff  
    inet 10.0.2.15/24 brd 10.0.2.255 scope global dynamic noprefixroute eth0  
        valid_lft 84917sec preferred_lft 84917sec  
    inet6 fe80::a00:27ff:fea6:1f86/64 scope link noprefixroute  
        valid_lft forever preferred_lft forever  
3: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 5  
    link/none  
    inet 10.41.0.66 peer 10.41.0.65/32 scope global tun0  
        valid_lft forever preferred_lft forever  
    inet6 fe80::e33b:472c:7b40:35a3/64 scope link stable-privacy  
        valid_lft forever preferred_lft forever
```

Wireshark



Herramienta de análisis de redes

- Permite analizar protocolos de red de las 4 capas
- También otros protocolos
 - USB
 - Bluetooth



Iniciar Wireshark

Grabar tráfico en una interfaz

Utilizar filtros

Filtros de visualización

https://www.wireshark.org/docs/wsug_html_chunked/ChWorkBuildDisplayFilterSection.html

— — —

- `udp,tcp`
 - `port`: puerto de origen o destino
 - `srcport`: puerto de origen
 - `dstport`: puerto de destino
- `ip`
 - `addr`: dirección de origen o destino
 - `src`: dirección de origen
 - `dst`: dirección de destino
- `comparadores y operadores`:
 - `comparadores`: `==`, `>`, `<`, `~`
 - `Operadores unarios`: `!`
 - `Operadores binarios`: `&&`, `||`, `^^`

Guardar captura

Interceptar tráfico cifrado

Hagamos un problema fácil

<https://2019shell1.picoctf.com/static/ae9ca8cff43ed638ed5d137f9ece7455/capture.pcap>

(PicoCTF 2019, Shark on Wire)

Otros protocolos interceptados

En CTFs a veces tienes que aprender cómo funcionan protocolo que nunca has visto para entender sus dump:

- USB (https://www.usb.org/sites/default/files/hut1_21.pdf)
 - Se pueden analizar paquetes del protocolo, para por ejemplo, determinar qué tecleó un teclado (Keyloggers) (Veremos esto en la auxiliar)
 - En casos en que OS no permita analizar paquetes, existen dispositivos *PITM* que se pueden conectar entre teclado y equipo y capturan lo transmitido.
- Bluetooth y otros protocolos inalámbricos
 - Se puede colocar algunas tarjetas de red en un modo especial que reciba e intercepte todos los paquetes a su alcance físico.