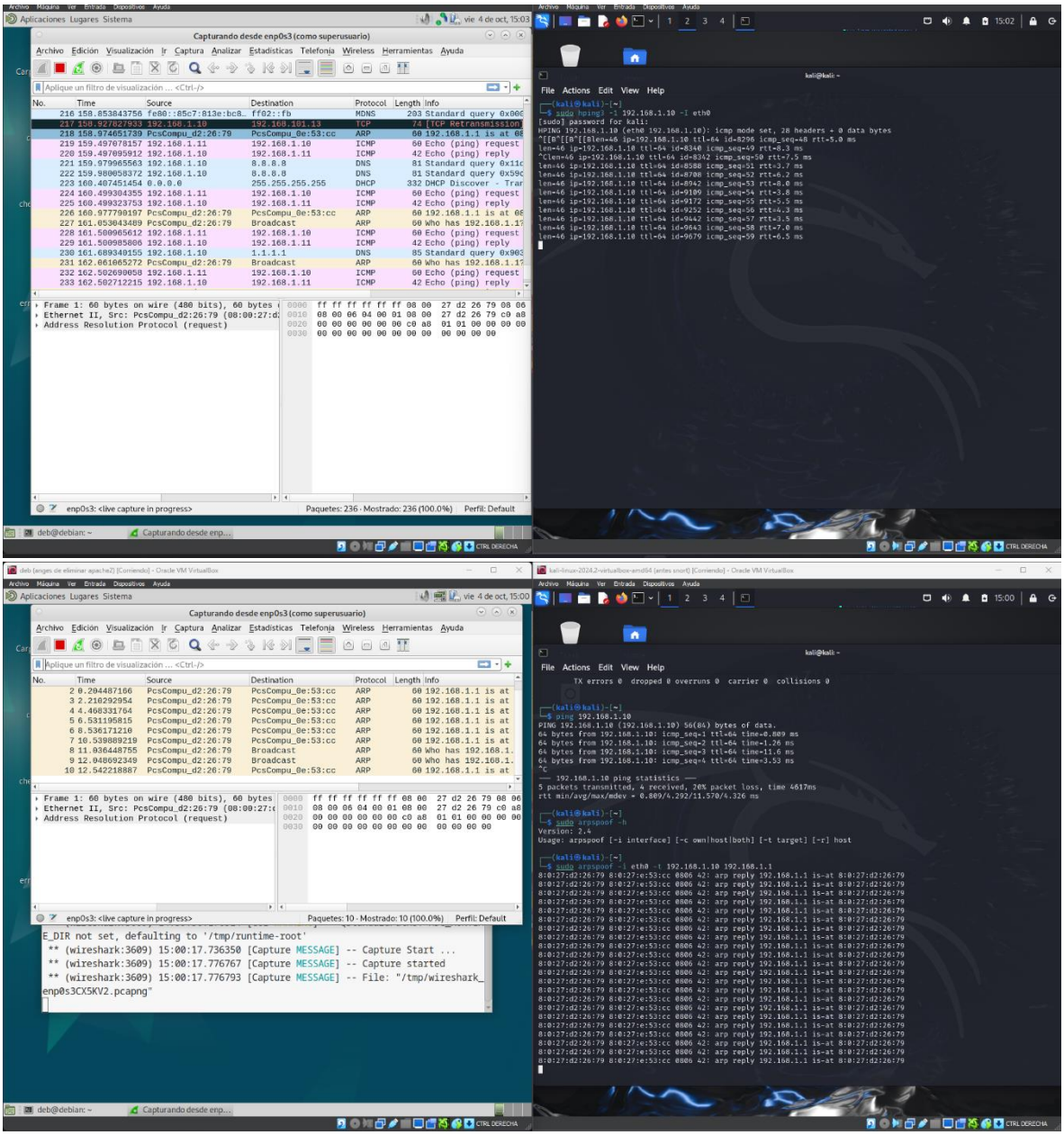
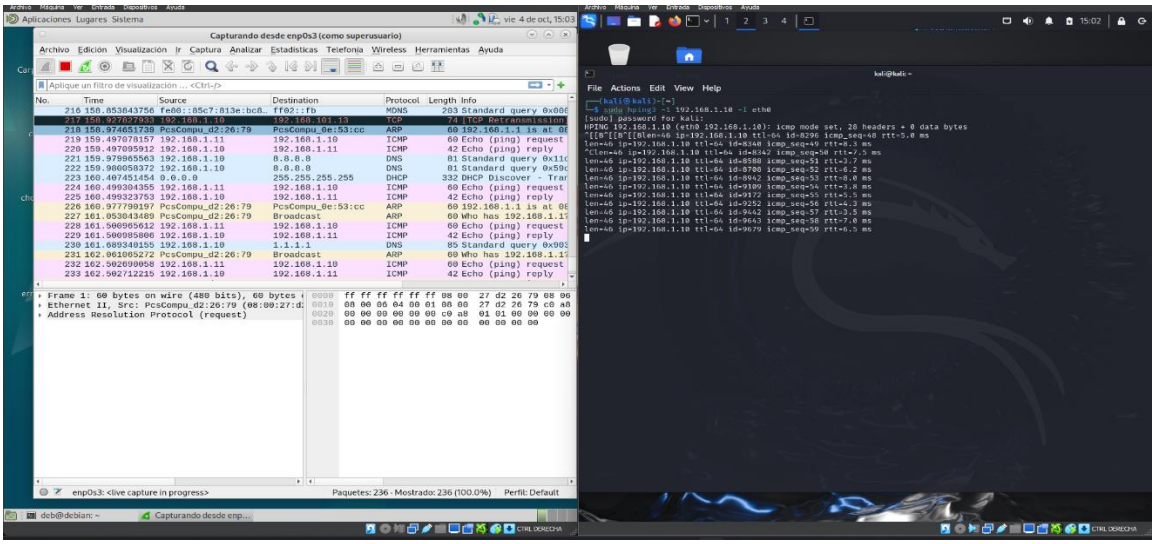
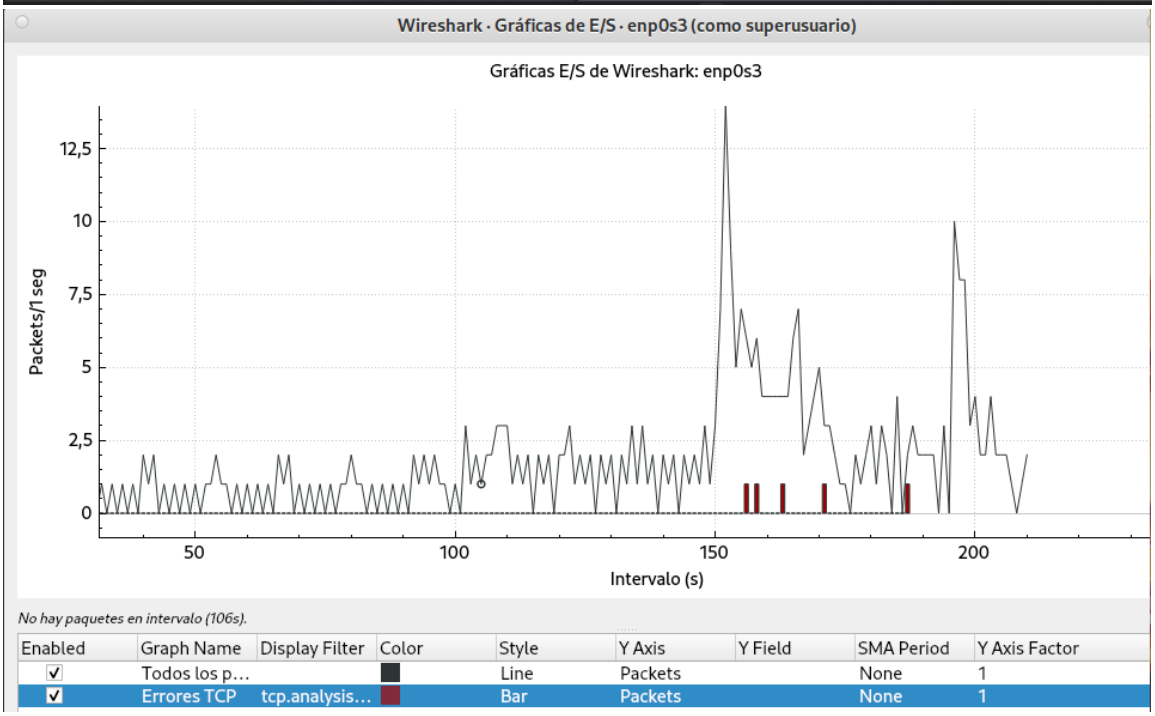
Ataque de Spoofing and DoS sobre un website WordPress








```
deb@debian:~/Escritorio$ cat detalles\ wirshark
# This file was created by Wireshark. Edit with care.
@Bad TCP@tcp.analysis.flags && !tcp.analysis.window_update && !tcp.analysis.keep_alive && !tcp.analysis.keep_alive_ack@[4626,10023,11822][63479,34695,34695]
@HSRP State Change@hsrp.state != 8 && hsrp.state != 16@[4626,10023,11822][65535,64764,40092]
@Spanning Tree Topology  Change@stp.type == 0x80@[4626,10023,11822][65535,64764,40092]
@OSPF State Change@ospf.msg != 10@[4626,10023,11822][65535,64764,40092]
@ICMP errors@icmp.type in { 3..5, 11 } || icmpv6.type in { 1..4 }@[4626,10023,11822][47031,63479,29812]
@ARP@arp@[64250,61680,55255][4626,10023,11822]
@ICMP@icmp || icmpv6@[64764,57568,65535][4626,10023,11822]
@TCP RST@tcp.flags.reset eq 1@[42148,0,0][65535,64764,40092]
@SCTP ABORT@sctp.chunk_type eq ABORT@[42148,0,0][65535,64764,40092]
@TTL low or unexpected@(ip.dst != 224.0.0.0/4 && ip.ttl < 5 && !pim && !ospf) || (ip.dst == 224.0.0.0/24 && ip.dst != 224.0.0.251 && ip.ttl != 1 && !(vrrp || carp))@[42148,0,0][60652,61680,
60395]
@Checksum Errors@eth.fcs.status=="Bad" || ip.checksum.status=="Bad" || tcp.checksum.status=="Bad" || udp.checksum.status=="Bad" || sctp.checksum.status=="Bad" || mstp.checksum.status=="Bad"
 || cdp.checksum.status=="Bad" || edp.checksum.status=="Bad" || wlan.fcs.status=="Bad" || stt.checksum.status=="Bad"@[4626,10023,11822][63479,34695,34695]
@SMB@smb || nbss || nbns || netbios@[65278,65535,53456][4626,10023,11822]
@HTTP@http || tcp.port == 80 || http2@[58596,65535,51143][4626,10023,11822]
@DCERPC@dcerpc@[51143,38807,65535][4626,10023,11822]
@Routing@hsrp || eigrp || ospf || bgp || cdp || vrrp || carp || gvrp || igmp || ismp@[65535,62451,54998][4626,10023,11822]
@TCP SYN/FIN@tcp.flags & 0x02 || tcp.flags.fin == 1@[41120,41120,41120][4626,10023,11822]
@TCP@tcp@[59367,59110,65535][4626,10023,11822]
@UDP@udp@[56026,61166,65535][4626,10023,11822]
@Broadcast@eth[0] & 1@[65535,65535,65535][47802,48573,46774]
@System Event@systemd_journal || sysdig@[59110,59110,59110][11565,28527,39578]
```

1. Monitoreo del tráfico de red con Wireshark (Máquina Debian)

a) En **Wireshark**, se están capturando paquetes ARP y ICMP (peticiones y respuestas de ping). El análisis está orientado a observar la resolución de direcciones (ARP) y la respuesta de las máquinas a peticiones ICMP (ping).

b) La gráfica de **Wireshark** muestra un análisis de tráfico TCP en términos de paquetes por segundo y errores detectados.

c) Esta herramienta es útil para monitorear cómo se comporta una red bajo un ataque ARP spoofing o DoS, ya que permite detectar flujos anormales de paquetes o intentos de suplantación de identidad en la red.

2. Ataque de ARP Spoofing con Kali Linux

- En la terminal de **Kali Linux**, se esta utilizando la herramienta **arpsoof** para generar tráfico ARP falso en la red. Estás haciendo que las máquinas dentro de la red crean que el atacante (Kali Linux) es en realidad el router o el destino legítimo (192.168.1.1).

3. Discusión sobre Estrategias de Mitigación

a) El uso de **Wireshark** y **ARP spoof** es excelente para aprender a detectar y mitigar ataques en redes locales.

b) Para un servidor de **WordPress** u otros servicios, las siguientes medidas podrían ser efectivas contra ataques ARP Spoofing y DoS:

  - **Firewalls de red**: Configurar reglas de filtrado de tráfico en el firewall para limitar el tráfico anormal y bloquear intentos de spoofing.
  - **ARP Inspection**: Implementar técnicas como **Dynamic ARP Inspection (DAI)** para validar las respuestas ARP y evitar ataques de suplantación.
  - **Limitar la tasa de paquetes ICMP**: Configurar firewalls o herramientas de mitigación para limitar las tasas de paquetes ICMP para evitar que se utilicen en ataques de denegación de servicio.
  - **IDS/IPS**: Un sistema de detección o prevención de intrusos (IDS/IPS) puede ayudar a identificar y bloquear intentos de ataques ARP Spoofing.

4. Conclusión: Buenas Prácticas para Proteger WordPress

- Mantener actualizado el software del servidor y del sitio WordPress.
- Implementar **WAF (Web Application Firewall)** y filtros contra ataques DoS.
- Configurar correctamente las reglas de firewall para bloquear tráfico malicioso y conexiones anormales.