Informacion deteccion tarjetas promiscuas

Ernesto Velasco Soler

# Funcionamiento general de una tarjeta en modo promiscuo

Para capturar el tráfico de la red a la que estamos conectados hemos de “situar” la tarjeta de red en modo promiscuo ¿Y qué es esto del modo promiscuo?, por definición todos los adaptadores de red reciben los paquetes que son para ellos, (filtran por IP), pero el colocar el adaptador de red en modo promiscuo hace que no filtre, y vea todo el tráfico que está en la red.

Los sniffers funcionan por una sencilla razón: muchos de los protocolos de acceso remoto a las máquinas se transmiten las claves de acceso como texto plano y, por lo tanto, capturando la información que se transmite por la red se puede obtener este tipo de información y el acceso ilegítimo a una determinada máquina.

La forma más inmediata de saber si un determinado adaptador de red está en un modo promiscuo es utilizar el programa ifconfig. Este programa permite configurar los adaptadores de red instalado en una determinada máquina y obtener información de esa configuración.

## Activar modo promiscuo en una tarjeta de red (Linux)

## Capturar tráfico que circula por una interfaz

TCPDUMP

tcpdump -i eth0 host 10.0.0.1 captura el tráfico proveniente de un host

## Detección de tarjetas promiscuas en sistemas

Técnicas de Detección

**• El test DNS**

En este método, la herramienta de detección en sí misma está en modo promiscuo.

Creamos numerosas conexiones TCP falsas en nuestro segmento de red, esperando

un sniffer pobremente escrito para atrapar estas conexiones y resolver la dirección

IP de los inexistentes hosts.

Algunos sniffers realizan búsquedas inversas DNS en los paquetes que capturan.

Cuando se realiza una búsqueda inversa DNS, una utilidad de detección de sniffers

"huele" la petición de las operaciones de búsqueda para ver si el objetivo es aquel

que realiza la petición del host inexistente.

**• El Test ICMP**

Ping de Latencia.

En este método, hacemos ping al blanco y anotamos el Round Trip Time (RTT,

retardo de ida y vuelta o tiempo de latencia).

Creamos centenares de falsas conexiones TCP en nuestro segmento de red en un

período de tiempo muy corto. Esperamos que el sniffer esté procesando estos

paquetes a razón de que el tiempo de latencia incremente.

Entonces hacemos ping otra vez, y comparamos el RTT esta vez con el de la primera vez. Después de una serie de tests y medias, podemos concluir o no si un sniffer está realmente funcionando en el objetivo o no.

**• El Test del Ping**

Este método confía en un problema en el núcleo de la máquina receptora. Podemos

construir una petición tipo "ICMP echo" con la dirección IP de la máquina

sospechosa de hospedar un sniffer, pero con una dirección MAC deliberadamente

errónea.

Enviamos un paquete "ICMP echo" al objetivo con la dirección IP correcta, pero con

una dirección de hardware de destino distinta.

La mayoría de los sistemas desatenderán este paquete ya que su dirección MAC es

incorrecta. Pero en algunos sistemas Linux, NetBSD y NT, puesto que el NIC está

en modo promiscuo, el sniffer asirá este paquete de la red como paquete legítimo y

responderá, por consiguiente. Si el blanco en cuestión responde a nuestra petición,

sabremos que está en modo promiscuo.

Un atacante avanzado puede poner al día sus sniffers para filtrar tales paquetes

para que parezca que el NIC no hubiera estado en modo promiscuo.

**• El test ARP**

Podemos enviar una petición ARP a nuestro objetivo con toda la información rápida

excepto con una dirección hardware de destino errónea.

Una máquina que no esté en modo promiscuo nunca verá este paquete, puesto que

no era destinado a ellos, por lo tanto, no contestará.

Si una máquina está en modo promiscuo, la petición ARP sería considerada y el

núcleo la procesaría y contestaría. Por la máquina que contesta, sabemos que

estamos en modo promiscuo.

**• El test Etherping**

Enviamos un "ping echo" al host a testear con una IP de destino correcta y

dirección MAC falseada.

Si el host responde, es que su interfaz está en modo promiscuo, es decir, existe un

sniffer a la escucha y activo.

* <https://pub.phyks.me/sdz/sdz/manipulez-les-paquets-reseau-avec-scapy.html>
* <https://stackoverflow.com/questions/44202550/python-3-mac-address-based-on-ip-scapy>

**Otras formas de Detectar Sniffers**

• Detectar y controlar los logs que suelen generar los sniffers.

• Detectar y controlar las conexiones al exterior.

• Monitorizar los programas que acceden al dispositivo de red.

• Normalmente una interface en modo promiscuo, queda reflejada en el

fichero de logs.

## Detección en Sistemas Windows

• PROMISDETECT

http://www.ntsecurity.nu/cgi-bin/do...scdetect.exe.pl

http://ntsecurity.nu/downloads/promiscdetect.exe

• ProDETECT 0.2 BETA

http://sourceforge.net/projects/prodetect/

http://prdownloads.sourceforge.net/prodetect/proi386.exe?download

Alerta de ProDETECT:

**Prueba de DNS :** muchas herramientas de rastreo de paquetes realizan una dirección IP para nombrar las búsquedas para proporcionar nombres DNS en lugar de direcciones IP. Para probar esto, debe colocar su tarjeta de red en modo promiscuo y enviar paquetes a la red destinada a hosts falsos. Si se observan las búsquedas de nombres de los hosts falsos, es posible que un rastreador esté en acción en el host que realiza las búsquedas.

**Prueba ARP :** cuando está en modo promiscuo, el controlador de la tarjeta de red verifica que la dirección MAC sea la de la tarjeta de red para paquetes de unidifusión, pero solo compara el primer octeto de la dirección MAC con el Valor 0xff para determinar si el paquete se transmite o no. Tenga en cuenta que la dirección de un paquete de difusión es ff: ff: ff: ff: ff: ff. Para probar esta falla, si envía un paquete con una dirección MAC de ff: 00: 00: 00: 00: 00 y la dirección IP de destino correcta del host. Después de recibir un paquete, el sistema operativo de Microsoft que usa el controlador defectuoso responderá mientras está en modo promiscuo. Probablemente ocurra solo con el controlador MS predeterminado.

**Ether Ping test :** en los kernels de Linux más antiguos, cuando una tarjeta de red se coloca en modo promiscuo, todos los paquetes pasan al SO. Algunos kernels de Linux solo miraron la dirección IP en los paquetes para determinar si deberían procesarse o no. Para probar esta falla, debe enviar un paquete con una dirección MAC falsa y una dirección IP válida. Los kernels de Linux vulnerables con sus tarjetas de red en modo promiscuo solo tienen en cuenta la dirección IP válida. Para obtener una respuesta, se envía un mensaje de solicitud de eco ICMP dentro del paquete falso que conduce a los hosts vulnerables en modo promiscuo para responder.

Quizás haya más, la prueba de DNS para mí es la más confiable

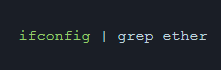
<https://hacking-etico.com/2010/08/26/hablemos-de-spoofing/>

Petición ARP : <https://es.slideshare.net/DiegoCaceres12/hack-x-crackscapy2-50858809>

Manipulación Avanzada Scapy: <https://seguridadyredes.wordpress.com/2009/12/03/scapy-manipulacion-avanzada-e-interactiva-de-paquetes-parte-1/>

Comprobación de Tarjeta promiscua: <https://aratech.es/redes/detectando-nics-en-modo-promiscuo>

## Mostrar MAC en Linux



## Mostrar MAC en Windows

Ipconfig /all

## Mostrar estructura protocolo

Ls (‘protocolo’)

## Cantidad de paquetes contestados

Len(ans) //Cuando utilizamos “ans,unans=srp(,timeout=2, retry=-3, iface=”interfaz”)

