ARP Spoofing Defense

Transcribed on August 2, 2025 at 1:16 PM by Minutes AI

Speaker 1 (00:02)

Bienvenidos a esta nueva sesión, en esta sesión vamos a tratar el tema de la resolución del ejercicio sobre mitigación de ataques IPV sobre varios ataques, en concreto era el ATP spoofing, pin float, etc.

Bien, pues vamos a proceder con la resolución de todas las soluciones para evitar los ataques que hemos comentado antes en el ejercicio anterior y comenzaremos por arpspoofing.

Bien, ya sabemos cómo funciona arpspoofing, lo que sí os cuento brevemente lo que vamos a hacer para protegerlo, y aquí hago un pequeño inciso, muchísimas formas de protegerlo, desde utilizar cortafuego, algún tipo de IPs, algún software específico para proteger del ATP spoofing, hay muchas formas, pero yo aquí me voy a centrar en una solución común que nos sirva en cualquier kernel de Linux y que nos sirva en cualquier máquina que tenga una clip de Linux que podamos utilizar como administrador, no utilizaré nada que sea comercial o un software, solamente algún tipo de aplicación que se puede instalar directamente en Linux en la cli en la línea de comandos.

Entonces centrándonos en el atlantizeproofing, lo primero que hay que hacer es la instalación de las herramientas de seguridad, en este caso utilizaremos dsniff que ya usamos en otros ejercicios, que es un paquete que incluye el arp spoof como ya sabéis, y este nos va a servir para detectar intentos de ARP spoofing en la red.

En este caso utilizaremos dsn iff que ya usamos en otros ejercicios, que es un paquete que incluye el arp spoof como ya sabéis, y este nos va a servir para detectar intentos de arp spoofing en la red.

Después haremos una configuración directa al kernel, porque vamos a aj configuraciones del kernel para que se mejoren las de sistema.

Después haremos una configuración directa al kernel, porque vamos a las configuraciones del kernel para que se mejoren la resistencia al sistema frente racks ARP y lo que va a hacer es ignorar respuestas ARP que no sean solicitadas y además también ajustando cómo se anuncia el ARP en la red.

Y al final, como ya sabéis en todo el ciclo de ciberseguridad no sólo es solucionarlo, sino cuando ya lo hemos solucionado tenemos que hacer un seguimiento, una monitorización y para eso utilizaremos arp watch que también veremos que nos sirve para monitorizar los cambios en las asignaciones de direcciones IP a direcciones Mac, y esto nos permite detectar cualquier tipo de actividad sospechosa en la red.

Bien, pues como hemos dicho, el primer paso será instalar todas las herramientas que nos hace falta, así que haremos un sudo apt install dsni Yo lo tengo instalado, con lo cual me va a decir que ya lo tengo, pero bueno, ese sería el proceso.

Y ahora vendría la segunda parte, que es cómo habilitar la protección contra ARP spoofing en el kernel.

Entonces en Linux se pueden habilitar algunas características de seguridad directamente contra el kernel, y en este caso lo que vamos a hacer es que ignore respuestas ARP que no sean solicitadas.

Entonces usaremos el siguiente comando.

Haremos un echo, este formato ya lo habéis visto en algunos otros ejercicios para asignar algún tipo de flag o de bit dentro de lo que es el núcleo de Linux, con lo cual haremos un net ipv, otro punto con otro all y arpignore es un comando que hay que meter con mucho cuidado para que no cometamos ningún error.

Después haremos un p y lo conectaremos con un sudo p, Ahora os contaré todo lo que hace cada uno de los comandos.

Y apuntamos al etc sysctl com Esto simplemente es, como ya habéis visto en otras ocasiones, asignarle algún valor dentro de ese fichero, algún flag concreto, en este caso el net ipv com all rtignore Entonces le damos y ya lo hemos fijado.

Y ahora repetimos igual, solo que cambiamos el ignore por anuncio, que justamente lo que hemos dicho antes es ignorar las respuestas no solicitadas, por ejemplo.

Entonces le damos aquí, le ponemos un 2 y hacemos todo lo demás se queda exactamente igual.

Ya le hemos puesto y era simplemente el servicio hay que reinicializarlo.

TTL p Vale, pues ahora.

Vale, pues ahora os explicaré un poco qué es lo que hemos hecho con este comando.

Bien, pues el flag este que he puesto tan largo de netipv unknown, que ha sido el segundo que he puesto, le hice al sistema que anuncie su propia dirección IP como la fuente de todas las respuestas ARP que envía.

Entonces esto nos ayuda Anuncie su propia dirección IP como la fuente de todas las respuestas ARP que envía.

Entonces esto nos ayuda a mitigar ataques donde un atacante intenta envenenar las tablas ARP de los dispositivos de la red, ya que los dispositivos legítimos pueden recibir respuestas ARP genuinas del sistema configurado con esta opción y la otra opción que era el 1, con el 1 le estamos indicando al sistema operativo que ignore los paquetes ARP que provienen de interface donde la dirección IP de destino no coincide con la dirección IP de ninguna interfaz local y esto nos ayuda a prevenir ataques donde un atacante envía

¿Veis el punto?

Estamos intentando decirle al kernel que no le haga caso a no ser que tenga una cierta configuración, sería la primera parte.

Como os dije antes también tenemos que monitorizar una vez hecho estos cambios tenemos que utilizar la aplicación que hemos dicho que era el arp watch, hacemos apt install arp watch y esta nos va a servir para monitorizar continuamente todo el tráfico que pasa por nuestra interfaz de red, entonces haríamos sudo a y aquí nuestra interfaz de red que en mi caso es la eth y esto ya empezaría a estar monitorizando los cambios en la dirección Mac y para ver esos cambios que se vayan ocurriendo haríamos un sudo grep y aquí estamos filtrando una cosa, si veis este mensaje como me ha pasado a mí, esto está indicando que el arp dat no existe, entonces depende de qué versión de Linux tengamos tenemos que crearlo ya que es importante porque es en este fichero el mismo sistema lleva un registro de las direcciones IP y Mac que son conocidas en la red, entonces puede ser que con esto provoque que no funcione.

Podemos crearlo nosotros mismos, como la carpeta bar lib en mi sistema ya existe, lo que voy a hacer es crearlo yo con un simple touch para que esté vacío pero que exista.

Var lib Aquí tenemos a arpwatch y le llamamos atc ya con esto lo creamos también conviene ponerle los permisos por si acaso hacemos un show en el que pongamos ardp watch que es el grupo, esta es una típica asignación de servicio para que no haya problemas libertad si todo está bien es y bien Y ya sólo quedaría reiniciar el servicio que hacemos con un sudo start system start y hacemos un ATP watch Ya lo tenemos, ahora volvemos a poner la salida del syslog está todo bien, en principio está Vale pues ya hemos solucionado este pequeño problema, si ahora se lanza otra vez el servicio de escucha, pues no debería de salirnos ese error más.

Vale, fijaros, voy a hacer para que veáis cómo funciona.

He hecho un cambio de la dirección Mac en la otra máquina, en este equipo he ido cambiando diferentes Mac, lo que hago es que apago el link, le meto una dirección nueva Mac, lo levanto otra vez, lo apago otra vez, voy haciendo cambio de Mac para que veáis cómo se registra en otro servidor.

Entonces con esto he hecho unos cambios que se deberían de reflejar, fijaros cómo es la dirección Mac acaba en la última o FF.

Si vuelvo otra vez al servidor y vemos el registro con el syslog, esos cambios están reflejados aquí.

Fijaos que te lo está diciendo una estación nueva en esta IP ha cambiado su dirección Mac a esta.

Y lo mismo con los tres cambios que he hecho yo hace un momento.

¿Ves qué gran utilidad es esta?

Cualquier cambio que haya en una Mac no lo va a reportar con ARP watch, con lo cual esto se puede concatenar por ejemplo con un correo electrónico o con cualquier cosa para que te alerte cada vez que se realiza un cambio en una dirección Mac de toda la red.

Bien, pues hasta aquí sería un poco un tipo de solución para el ARP Spoofing, ya os digo, hay muchas, pero con esta por lo menos tenemos una base en la que podemos además de solucionar ciertos problemas del ARP Spoofing, también monitorizarlo.

Con lo cual pasamos al siguiente ejercicio, vamos a nivelar las máquinas.

Bien, ya he reiniciado las máquinas a un estado limpio para que no haya conflictos y empezamos con el siguiente ejercicio que era el ping float.

El ping float es un tipo de ataque que ya sabéis que es denegación de servicio y que lo que hace es sobrecargar la del sistema.

Competiciones ICMP, de ICO request, de ECO request, que es un SEL pin.

¿Qué es lo que vamos a hacer para solucionarlo?

Bien, pues lo primero será hacer una limitación de peticiones ICMP con iptable.

Entonces aplicaremos estas reglas indipetables, que ya lo conocemos, para limitar la tasa de ping entrante.

Esto permite que si el sistema detecta algún problema sea capaz de ignorar cuando haya muchas peticiones y potencialmente que son un problema.

Y después utilizaremos la protección Sync proxy, le pondremos una regla adicional en iptables para activar esta protección que ya os digo, se llama Synproxy y ayuda a mitigar ataques más complejos, más sofisticados.

Esta solución es un poco más técnica y lo que hace es proteger contra, por ejemplo, la saturación de conexiones TCP.

Y bueno, aquí no aplicaré ninguna herramienta para monitorizarlo porque el mismo iptable sabéis que tiene un log y podemos ver perfectamente todo lo que está ocurriendo, así que bueno, vamos a implementar la solución.

Bien, pues voy a proceder como he dicho a introducir esas reglas.

Para ello realizaremos ip tables como ya conocemos bastante bien, pondremos un guión a como siempre sabéis que es el input ICMP, ahora os cuento lo que hace type, bueno, conocéis más o menos ya todos los comandos, de hecho hicimos uno muy parecido a este, echo echo request le ponemos un limit y después le ponemos limit, de hecho se parece muchísimo al que ya hicimos en otro ejercicio, accept.

Bien, ¿Qué es lo que hemos hecho aquí?

Bueno ya conocéis el AINPUT, el PICMP que es el protocolo y después la cadena ICMP type echo request se refiere al tipo de mensaje ICMP, en este caso como ya sabéis es el icorequest que es el PIN, Después el limit limit s lo que hace es utilizar el modo limit para establecer un límite de velocidad en un paquete por segundo para los paquetes ICMP.

Y ya finalmente el acept, si se cumplen las condiciones anteriores se acepta el paquete, ponemos la suite y seguimos.

El siguiente comando es este, haremos un sudo iptables en el que pondremos ip tables guión a input haremos el input guión p icmp como arriba icmp hasta aquí icmp hasta aquí está bien, ICMP type echo request j crop Vale, esto es lo contrario, entonces lo que estamos haciendo es justamente es parecida a la primera regla como veis, pero esto agrega otra regla al input para el tráfico ICMP del eco request y es el drop, en este caso si encuentro un paquete ICMP del tipo echo request directamente se descarta sin responder.

Vale, eso es lo que hemos hecho, le damos y lo aplicamos y lo aplicamos a verlo.

ICMP type Si vale, tanto comandos son tan complejos que hay que tener mucho cuidado.

Os dejo el error este para que veáis lo fácil que es cometer algún fallo, ha sido por esto ICMP y ahora ya ponemos request, esta vez sí fuera justamente lo que os he dicho es lo que hemos aplicado con estas dos reglas.

Bien, ¿Cuál será el siguiente paso?

Pues el siguiente paso que tenemos era el que comentamos antes que era habilitar el SIM proxy, entonces aquí ya hay que meter una cantidad de comandos un poco más complejo, lo pongo todo el tirón para no ir escribiendo y para que veáis un poco lo que hemos hecho, bien, para no teclear todos estos comandos, como podéis ver es muy complejo, pero esto lo que sería aplicar lo que he dicho antes del Simbrox.

Bien, ¿Que es lo que hace cada uno de ellos?

Fijaros en la primera, la primera línea que veis aquí, voy a contaros qué hace esta línea, esa línea lo que hace con el guion T, bueno impetable, especifica la tabla RAW, esta se utiliza fundamentalmente para configurar reglas antes de que se haga ningún seguimiento de conexión.

Después el PREROUTING, lo que hace es que agrega una regla de la cadena PREROUTING que controla el procesamiento de los paquetes antes de ser enrutado, esto lo conocemos ya lo hemos utilizado antes y bueno lo siguiente es P MTCP SYN, aquí se especifica que la regla se va a aplicar a los paquetes TCP con el FLAG SYN, recordad que esto era la petición de inicio de conexión y después el JCT NO TRACK, esta acción es para el paquete que coincide con la regla, esto se aplica al paquete que coincide con la regla, en este caso se pasa al módulo CT que es el contract para no rastrear el estado de la conexión, es un poco lo que hace esta cadena.

La siguiente, la de abajo, aquí se aplica tanto al tráfico TCP entrante que tiene el FLAG SYNC activo y establece el SYNC PROXY, que es la técnica que hemos dicho que es de protección contra ataques de inundación SYNC, el TCP FLAG FIN SYNC, ACK SYNC, todo eso especifica que el paquete debe tener el FLAG SYNC activo y no tener ningún otro más solamente.

Y después J SIMPROXY, es ya donde aplicamos el SYNPROXY a los paquetes que cumplen con las condiciones que hemos especificado.

Y bueno sac per timestab scale 7, bla bla, 1040, 1004, estos son parámetros adicionales que se pasan a simproxy para configurar la protección, configuran opciones como el TCP, como el permiso de SAC SACK que es SELECTIVE ANOLEX, la marca de tiempo, también la escala de la ventana y el tamaño máximo de segmento esto ya un poco técnico, aquí hay que conocer muy bien cómo funcionan las tramas de paquetes a un muy bajo nivel, pero bueno, yo lo dejo aquí por si queréis investigar cómo funciona y además os lo he puesto para que veáis la forma tan enorme que tenemos de personalizarlo hasta este nivel de protección.

Y bien, el último comando que veis aquí con iptables lo que está haciendo es que se aplica a todos los paquetes entrantes que están en un estado de invalid, como podéis ver aquí, y el state invalid lo que hace es que hace coincidir los paquetes que tienen un estado de conexión inválido, es un poco eso y nada, y el drop ya sabéis lo que hace, que directamente descarta los paquetes que cumplen con la condición del estado inválido.

Bien, pues esto sería un poco la activación de estas reglas de hipetables, están diseñadas para proteger contra muchos tipos de ataques de red, como inundaciones sin como el Pink Flute que estamos trabajando contra él, y también lo que hace es evitar el seguimiento de conexiones para cierto tipo de paquetes, y esto ayuda a prevenir muchos ataques y también mejora mucho la seguridad de nuestro servidor.

Así que bueno, hasta aquí un poco la defensa que podemos aplicar con Invitables para el pin float.

Bien, vamos a ver ahora cómo implementar algunas soluciones para evitar justamente el ataque DHCP Starvation o agotamiento de direcciones IP a un servidor DHCP.

Antes comentamos que una de las soluciones era activar el DHCP Snooping, eso lo podemos hacer en función de qué tipo de hardware tengamos, si tuviéramos equipos Cisco podríamos utilizar comandos directamente, como por ejemplo el que podéis ver aquí, aunque esto es de Cisco, lo escribo aquí, este comando por ejemplo lo que haría sería activar el snooping directamente, el DHCP snooping directamente contra el switch, ya sólo nos quedaría decir en que VLAN, en qué puerto, etc.

Recordad, esto es un comando de Cisco, os lo he puesto aquí para que lo sepáis cómo se podría hacer directamente contra el Cisco iOS.

Bien, pues centrándonos ahora ya en lo que podemos hacer contra el servidor, recordar que muchas de las soluciones contra los ataques asociados a DHCP hay que implementarlos desde la red y no desde el servidor, pero bueno, daré algunas pautas para seguir que podemos aplicar en el mismo servidor, implementarlos desde la red y no desde el servidor, pero bueno, os daré algunas pautas para seguir que podemos aplicar en el mismo servidor y muchas de las claves pasan por editar el fichero conf que ya conocemos.

Por ejemplo, si editamos el fichero etcp que ya conocemos, que hemos visto antes, aquí podemos añadir lo siguiente.

Aquí podemos ver la configuración que utilizamos al hacer la prueba.

Entonces aquí lo que podemos es añadir algún host conocido, información de los host conocidos y así podemos reservar la dirección IP para ese dispositivo.

Esto es muy bueno, muy práctico para lo que ya os he dicho antes, para aquellos equipos críticos, aunque se recomienda mucho más hacerlo de forma manual, pero también podemos tener cierta seguridad si hacemos una reserva IP para aquellos equipos no tan críticos que están en un punto intermedio, como por ejemplo ciertos usuarios.

Y eso lo podríamos hacer añadiendo lo que acabo de poner aquí debajo en la configuración.

Como podéis ver, lo que está diciendo es que la Mac, que la AABBC, la de ejemplo, es la dirección Mac del dispositivo y la dirección IP que acaba en 100 es la dirección IP que queremos reservar para que se le asigne a ese dispositivo.

Pues bien, para equipos críticos, equipos que queramos controlar de una forma específica, podemos hacer esta técnica, con lo cual no habrá nunca un problema de asignarle una IP a esa máquina, porque ya la tiene asignada desde el momento que se levanta el servidor.

Bien, pues volviendo a editar otra vez el fichero com, esta vez lo que vamos a aplicar es la limitación de las peticiones DHCP.

Esto es crítico, es un poco complejo de dar con la clave, de buscar los números exactos y se ajustan directamente aquí en las dos líneas que podéis ver aquí abajo.

Estas dos líneas son las que asignan esos tiempos, que ya lo hicimos también nosotros aquí, si os acordáis cuando lo configuramos también las pusimos por defecto aquí, pero las que están de forma genérica son las que habéis visto arriba.

Estas serían las que asignamos para esta subnet, con lo cual nos permite incluso esa segmentación, poner un default, uno por defecto y después asignar esos tiempos también de forma independiente según la la subnet que queramos gestionar.

Y como siempre, también tenemos que monitorizar todos los eventos que han ido ocurriendo asociados al DHCP.

Para ello podemos utilizar el siguiente comando, que es el comando tail con el guión f en el que ponemos var y nos syslog creo.

Con esto vamos a mirar al registro general de eventos de log que ya sabéis cualquier syslog y de ahí pues veremos con un grep apuntando solamente a aquellos asociados con el DHCP, en este caso lo hemos puesto en tiempo real, fijaos que el cursor se ha quedado aquí pendiente, con lo cual ahora iría registrando en tiempo real cualquier entrada que esté relacionada con DHCP, iremos viendo el log y a medida que entre algo o una operación instalada asociada o aparezca el término DHCP va a aparecer aquí, fijaos he vuelto a lanzar el ataque que hicimos en el anterior ejercicio y ya podéis ver cómo en tiempo real va registrando todo lo que está ocurriendo en los log, veis que ya no hay free lease y con lo cual ha entrado en ese bucle de no asignar más direcciones IP, con lo cual esto es una buena opción, teniéndolo un terminal aparte podemos ir trabajando con él y ahí iremos viendo las diferentes evoluciones que están asociadas siempre con el DHCP en tiempo real.

Bien y ya por último pasamos al DNS spoofing, para ello utilizaremos o vamos a configurar DNSSEC para validar las respuestas DNS y así asegurarnos que la información que recibimos es auténtica, además utilizaremos resolver DNS o una serie de resolvers de DNS que soporten DNSSEC para agregar una capa adicional de seguridad en la resolución de nombres de dominio.

Bien pues el DNS Security Extension que es el DNSSEC nos ayuda a proteger contra el DNS spoofing validando las respuestas DNS, con lo cual haremos un sudo apt install el bind 9 que es el paquete que lleva esa gente que lleva esa aplicación, instalamos, ya lo tenemos, después tenemos que usar esta serie de comandos para poder activarlo que es el confianza, después tenemos que crear otra con named checkconf y finalmente un sudo system Entonces lo que os decía de los resolvers de los resolvers o los resolver es que traducirlo complicado pero los resolver seguros para configurarlo podemos hacer esto, podemos hacer un echo nameserver y ahora ponemos sudotti, vamos a almacenarlos en el fichero de configuración que ya conocemos de la TNS que es y bien de esta forma lo que haremos será ir añadiendo aquellos servidores que acepten DNSSEC al fichero conf, pues bien, comprender y aplicar medidas de mitigación contraataque son fundamentales para garantizar la seguridad de los sistemas y las redes informáticas.

Y desde un punto de vista práctico, estas habilidades son esenciales para proteger lo que ya conocemos, la integridad, la disponibilidad y la confidencialidad de la información frente a las constantes amenazas que tenemos en la ciberseguridad.

Llegamos al final de la sesión, os esperamos en el siguiente vídeo.