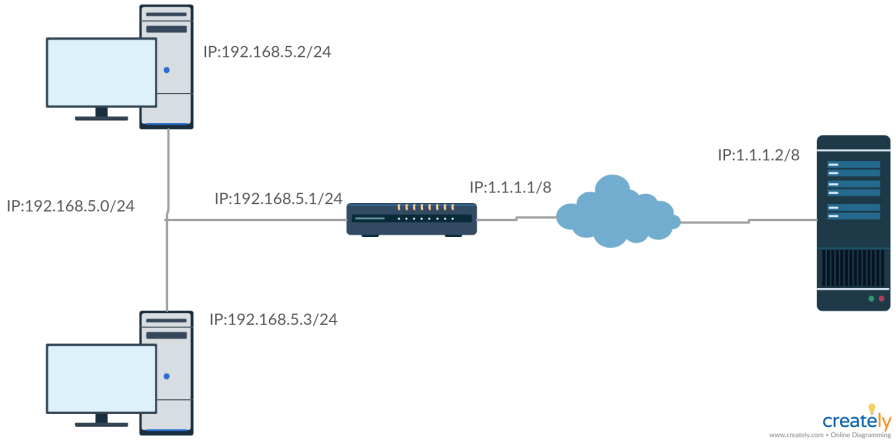
Network Address Translation

Antunez Joaquin, Gonzalez Alejo, Nielsen Maximiliano

Junio 2019

Somos el encargado de redes en la empresa “noQueremosLaburarAsiQueRobamosCodigo”. En esta empresa contamos únicamente con 2 equipos, uno que corresponde al pasante de la empresa y otro a Martín el desarrollador principal. En esta empresa contamos con un servicio especial por parte de “StackNotInOverflow” que nos provee con código de asistencia para el desarrollo de nuestra aplicación. “StackNotInOverflow” nos provee el código mediante una página web que está hosteada en un servidor apache.
Los hosts de nuestra red poseen todas ip privadas, mientras que el servidor de “StackNotInOverflow” está bajo una ip pública. Este es el gráfico de nuestra red:



El host 2 de la red privada corresponde a Martín y el 3 al pasante.

INTRODUCCIONTEORICADENAT

Introducción Lab Nat con IPtables

¿Qué es NAT?

El proceso de Network Address Translation es un mecanismo usado por los routers IP para que dos redes puedan intercambiar paquetes aunque tengan direcciones incompatibles. En una estructura donde varios hosts tienen direcciones IP privadas (red interna), cuando estos quieren enviar paquetes fuera del router NAT se encarga de traducir esa dirección interna (de rango privado) en una externa (de rango público). Cuando se creo el Internet en el año 1969, no se lo pensó con la magnitud que hoy tiene. El protocolo IPv4 consta 32 bits y, a día de hoy, un número limitado de direcciones IP; por eso es tan necesaria la NAT. Gracias a la misma se logra que, por ejemplo, en una red de una empresa

donde hay cientos de computadoras, se arme una red interna donde cada host tiene una dirección interna (privada) y así se tenga solo una dirección pública o a lo sumo unas pocas más, en vez de tener cientos de direcciones públicas. Esto es fundamental para el ahorro de direcciones IP públicas IPv4 ya que estas, como todos sabemos, no son infinitas y en algún momento se van a agotar.
Los usuarios internos utilizan normalmente la Source NAT para acceder a Internet; la dirección de origen se traduce y por lo tanto se mantiene privada. El NAT de destino se realiza en los paquetes entrantes cuando el firewall traduce una dirección de destino a una dirección de destino diferente; por ejemplo, traduce una dirección de destino pública a una dirección de destino privada.

Tipos de NAT.

Basic Nat:
Es el tipo más simple de NAT, provee una traducción de IP de tipo uno a uno. También es llamado como One-to-one nat. En este tipo de NAT solo la dirección IP, IP header checksum y cualquier otro checksum de más alto nivel que incluya la dirección IP son modificados. Este tipo de NAT puede usarse para conectar 2 redes IP que tengan direcciones incompatibles

One-to-many NAT:
La mayoría de NATs mapea múltiples host privados a una direccion IP publica. En una configuración típica, una red local usa una de las direcciones IP privadas designadas. El router esta también tiene una dirección de la red privada pero también está conectado a internet con una dirección pública. A medida que viaja información desde la red local a internet, la dirección de cada paquete es traducida de una dirección privada a la dirección pública. El router mantiene información básica sobre cada conexión activa. cuando una respuesta vuelve al router este usa la información de la coneccion, que mantuvo desde que se comenzó la coneccion, para enviar la respuesta a la dirección privada correcta

Métodos de traducción.

Full-cone NAT:

- Cuando una dirección interna (iAddr:iPort) es mapeada a una dirección externa (eAddr:ePort), cualquier paquete de iAddr:iPort es enviado a eAddr:ePort

- Cualquier host puede enviar paquetes a iAddr:iPort enviando paquetes a eAddr:ePort

(Address)-restricted-cone NAT:

- Cuando una dirección interna (iAddr:iPort) es mapeada a una dirección externa (eAddr:ePort), cualquier paquete de iAddr:iPort es enviado a eAddr:ePort

- Un host externo (*hAddr:any*) puede enviar paquetes a iAddr:iPort enviando paquetes a eAddr:ePort solo si iAddr:iPort este envió previamente un paquete a hAddr:*any*

Port-restricted cone NAT:

Es como Address-restricted-cone NAT pero además restringe el puerto.

- Cuando una dirección interna (iAddr:iPort) es mapeada a una dirección externa (eAddr:ePort), cualquier paquete de iAddr:iPort es enviado a eAddr:ePort
- Un host externo (*hAddr:hPort*) puede enviar paquetes a iAddr:iPort enviando paquetes a eAddr:ePort solo si iAddr:iPort este envió previamente un paquete a hAddr:*hPort*

Symmetric NAT:

- Cada solicitud desde la misma dirección IP interna y puerto a una dirección IP y puerto de destino específicos se asigna a una única dirección IP externa de origen y puerto; Si el mismo host interno envía un paquete incluso con la misma dirección y puerto de origen pero a un destino diferente, se utiliza una asignación diferente.

- Solo un host externo que recibe un paquete de un host interno puede devolver un paquete.

NAT está soportado por iptables mediante la tabla llamada nat y los objetivos SNAT, DNAT y MASQUERADE.
SNAT (Source NAT) implementa la traducción de la dirección de origen de paquetes que salientes (y la transformación correspondiente del tráfico de respuesta).
DNAT (Destination NAT) implementa la traducción de la dirección de destino de los paquetes entrantes (y la transformación correspondiente del tráfico de respuesta).
MASQUERADE is similar a SNAT, pero se usa en situaciones donde la dirección objetivo de la traducción es desconocida (ej: cuando se usa DHCP).

Para poder conectarnos al servidor tenes que habilitar la conexión usando DNAT.

iptables -t nat -A PREROUTING -p tcp -d 192.168.5.1 –dport 80 -j DNAT –to-destination 1.1.1.2:80

Este comando se traduce los intento de conexión TCP al puerto 80 en 192.168.5.1 al puerto 80 al puerto 80del host 1.1.1.2. También necesitaremos usar SNAT:

iptables -t nat -A POSTROUTING -s 192.168.5.0/24 -j SNAT –to-source 1.1.1.1

Esta cadena permite que el tráfico que tenga origen enla red 192.168.5.0/24 se lo tradusca a tener origen en 1.1.1.1 (la direccion publica del router).
Para poder ver las cadenas que añadimos usaremos el comando:

iptables -t nat -L

Ahora solo nos queda iniciar el servidor apache:

/etc/init.d/apache2 start

Y ahora pondremos a monitorear al acceso al servidor:

tail -f /var/log/apache2/access.log

Ahora intentaremos conectarnos al servidor mediante la PC1:

links 1.1.1.2

Tendríamos que ver “It works!”.

Introducción Firewalls

¿Qué es Firewall?

Firewall es un sistema de seguridad de red que monitorea y controla el tráfico de red entrante y saliente según reglas de seguridad predeterminadas. Un firewall generalmente establece una barrera entre una red interna confiable y una red externa no confiable, como Internet

Tipos de firewalls

Los firewalls generalmente se clasifican como basados en red o en host. los firewalls basados en red se ubican en las gateways de LAN, WAN e intranets. Los firewalls basados en host se ubican en el nodo de la red y controlan el tráfico de red dentro y fuera de las máquinas

Los firewall también varían en tipo dependiendo de dónde se origina la comunicación, dónde se intercepta y el estado de la comunicación que se está rastreando.

Network layer o packet filters:

Opera a un nivel relativamente bajo de la pila de protocolos TCP / IP, no permitiendo que los paquetes pasen a través del firewall a menos que coincidan con el conjunto de reglas establecido.

Application-layer:

Funciona en el nivel de aplicación de la pila TCP / IP (es decir, todo el tráfico del navegador, o todo el tráfico de telnet o FTP), y puede interceptar todos los paquetes que viajan hacia o desde una aplicación.

Los firewalls de aplicación funcionan determinando si un proceso debe aceptar alguna conexión dada. Los firewalls de aplicaciones cumplen su función

conectándose a las llamadas de socket para filtrar las conexiones entre la capa de aplicación y las capas inferiores del modelo OSI

Proxies:

Un servidor proxy (que se ejecuta en hardware dedicado o como software en una máquina de propósito general) puede actuar como un servidor de seguridad respondiendo a los paquetes de entrada (solicitudes de conexión, por ejemplo) en la forma de una aplicación, mientras bloquea otros paquetes. Un servidor proxy es una puerta de enlace de una red a otra para una aplicación de red específica, en el sentido de que funciona como un proxy en nombre del usuario de la red.

Network address translation:

Los firewalls a menudo tienen la funcionalidad de traducción de direcciones de red (NAT), y los hosts protegidos detrás de un firewall comúnmente tienen direcciones en el "rango de direcciones privadas". Los firewalls a menudo tienen dicha funcionalidad para ocultar la verdadera dirección de la computadora que está conectada a la red.

Implementacion Firewalls

Permitir conexiones entrantes establecidas y relacionadas

Como el tráfico de red generalmente necesita ser bidireccional (entrante y saliente) para funcionar correctamente, es típico crear una regla de firewall que permita el tráfico entrante establecido y relacionado, de modo que el router permita el tráfico de retorno a las conexiones salientes iniciadas.

iptables -A INPUT -m conntrack –ctstate ESTABLISHED,RELATED -j ACCEPT

Permitir conexiones salientes establecidas

Para permitir el tráfico saliente de todas las conexiones establecidas, que suelen ser la respuesta a las conexiones entrantes legítimas.

iptables -A OUTPUT -m conntrack –ctstate ESTABLISHED -j ACCEPT

Interno a Externo

eth1 es la red externa, y eth0 es la red interna, esto permitirá a la interna acceder a la externa:

iptables -A FORWARD -i eth1 -o eth0 -j ACCEPT

Soltar paquetes no válidos

Algunos paquetes de tráfico de red se marcan como no válidos. A veces puede ser útil registrar este tipo de paquete, pero a menudo está bien dejarlos caer.

iptables -A INPUT -m conntrack –ctstate INVALID -j DROP

Permitir todo SSH entrante

Para permitir que todas las conexiones SSH entrantes ejecute estos comandos:

iptables -A INPUT -p tcp –dport 22 -m conntrack –ctstate NEW,ESTABLISHED -j ACCEPT

Permitir Rsync entrante

Rsync, que se ejecuta en el puerto 873, se puede usar para transferir archivos de una computadora a otra.

iptables -A INPUT -p tcp –dport 873 -m conntrack –ctstate NEW,ESTABLISHED -j ACCEPT

Servidor Web

Los servidores web, como Apache y Nginx, normalmente escuchan las solicitudes en los puertos 80 y 443 para las conexiones HTTP y HTTPS, respectivamente.

Permitir todos los HTTP entrantes

Para permitir que todas las conexiones HTTP entrantes (puerto 80) ejecute estos comandos:

iptables -A INPUT -p tcp –dport 80 -m conntrack –ctstate NEW,ESTABLISHED -j ACCEPT

Permitir todos los HTTPS entrantes

Para permitir que todas las conexiones HTTPS (puerto 443) entrantes ejecute estos comandos:

iptables -A INPUT -p tcp –dport 443 -m conntrack –ctstate NEW,ESTABLISHED -j ACCEPT

Permitir todos los HTTP y HTTPS entrantes

Para permitir que todas las conexiones entrantes HTTP y HTTPS (puerto 443) ejecuten estos comandos:

iptables -A INPUT -p tcp -m multiport –dports 80,443 -m conntrack –ctstate NEW,ESTABLISHED -j ACCEPT

Mail

Los servidores de correo, como Sendmail y Postfix, escuchan en una variedad de puertos dependiendo de los protocolos que se utilizan para la entrega de correo.

Permitir todo SMTP entrante

Para permitir que su servidor responda a las conexiones SMTP, puerto 25, ejecute estos comandos:

iptables -A INPUT -p tcp –dport 25 -m conntrack –ctstate NEW,ESTABLISHED -j ACCEPT

Permitir todos los IMAP entrantes

Para permitir que su servidor responda a las conexiones IMAP, puerto 143, ejecute estos comandos:

iptables -A INPUT -p tcp –dport 143 -m conntrack –ctstate NEW,ESTABLISHED -j ACCEPT

Permitir todos los IMAPS entrantes

Para permitir que su servidor responda a las conexiones IMAPS, puerto 993, ejecute estos comandos:

iptables -A INPUT -p tcp –dport 993 -m conntrack –ctstate NEW,ESTABLISHED -j ACCEPT

Permitir todos los POP3 entrantes

Para permitir que su servidor responda a las conexiones POP3, puerto 110, ejecute estos comandos:

iptables -A INPUT -p tcp –dport 110 -m conntrack –ctstate NEW,ESTABLISHED -j ACCEPT

Permitir todos los POP3S entrantes

Para permitir que su servidor responda a las conexiones POP3S, puerto 995, ejecute estos comandos:

iptables -A INPUT -p tcp –dport 995 -m conntrack –ctstate NEW,ESTABLISHED -j ACCEPT

Bloquear todo o que no cumpla con las reglas establecidas

iptables -A INPUT -j DROP

Conclusión

Eso debería cubrir muchos de los comandos que se usan comúnmente al configurar un firewall de iptables. Por supuesto, iptables es una herramienta muy flexible, así que siéntase libre de mezclar y combinar los comandos con diferentes opciones para satisfacer sus necesidades específicas si no están cubiertas aquí.