Redes
Control 3

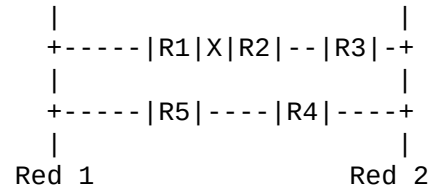
José M. Piquer, N. Becerra - 2 hrs
Con Apuntes - Hojas Separadas

22 de noviembre de 2010

Pregunta 1 (Ruteo)

Parte I

Muestre un ejemplo en que Split Horizon con Poison Reverse permiten evitar que se forme un ciclo en las tablas de rutas.



Si se cae el enlace entre R1 y R2, R2 borra de su tabla de rutas el camino a Red1. Entonces recibe desde R3 la misma entrada que él había avisado antes. Con Split Horizon y Poison Reverse, recibe Net1 con distancia infinito, porque R2 sabe que recibió esa ruta por la misma interfaz que está usando para anunciarla, por lo que ahora no se genera el loop.

Parte II

Argumente en qué tipos de redes ud recomendaría RIP y en cuales OSPF. Explique ppor qué.

RIP es un buen protocolo para usar en redes simples, no demasiado grandes y donde no haya una administración central clara. OSPF es mejor en redes más grandes, donde haya varios administradores distribuidos y autónomos, y tengamos capacidad de ingeniería para hacer el diseño y configuración de las áreas.

Parte III

En un momento ocurre que un Sistema Autónomo queda dividido en dos islas no conectadas directamente, cada una con varias redes IP y conectadas a Internet. Explique si BGP4 tolera esto y por qué.

Un Sistema Autónomo no se puede dividir en dos. Si eso ocurre, la red queda particionada, en algunas partes de Internet veo una parte y la otra no, y en otras lo contrario.

Pregunta 2 (Firewalls y DNS)

Parte I

Los promotores de IPv6 argumentan que NAT ya no será necesario y que era tremendamente dañino para Internet, al romper el principio de conectividad total. Los administradores de redes, en cambio, piensan que NAT aumenta la seguridad de sus redes y creen que debiera utilizarse también en IPv6. ¿Quién tiene razón? Argumente técnicamente.

argumentos en contra de NAT: efectivamente rompe la conectividad global, ya no se puede accesar a una máquina desde cualquier otra, obligando a establecer protocolos complejos de NAT inverso o proxies. La seguridad de esos protocolos es dudosa, puesto que es fácil olvidar las excepciones que alguna vez se agregaron.

argumentos a favor de NAT: al no ser accesibles las máquinas desde Internet, el control del acceso y la seguridad se hacen más fáciles, no se requiere filtrar en base a reglas, simplemente no son accesibles. Si la gran mayoría de las máquinas internas no serán accesibles desde Internet, puede ser útil.

Parte II

Una desventajas de tener un firewall tipo proxy es que debemos tener un proxy para cada protocolo/aplicación que desarrollemos. ¿Es posible tener un proxy genérico, que funcione para todo protocolo/aplicación?

Si, es posible, pero debemos interceptar el paquete de conexión TCP y lograr que el proxy realice la conexión correspondiente y luego traducir los puertos utilizados. Si no tenemos acceso a la capa TCP, y es simplemente una aplicación, no es posible hacer un proxy genérico porque no sabemos a qué máquina ni a qué port se quiere conectar realmente. Pero podemos instalar un proxy genérico que simule un servidor dado, redirigiendo las conexiones a otro servidor. Por supuesto que funciona solo para TCP y para protocolos básicos. Si el servidor se conecta de vuelta para verificar, o utiliza otras conexiones, no funcionará.

Parte III

El DNS utiliza una lista de servidores para cada dominio. Al tener que resolver una pregunta bajo ese dominio, siempre existe la duda de ¿a cual servidor de la lista le pregunto? Proponga un algoritmo que resuelva esa pregunta tratando de minimizar los tiempos de respuesta y de ser adaptivo al mismo tiempo.

- si nunca he usado ninguno, busco uno al azar y le pregunto.
- si he usado algunos, busco el que respondió más rápido
- al obtener respuesta: anoto el tiempo de respuesta obtenido para este servidor
- si pasa más de 3 segundos sin respuesta, vuelvo a buscar
- si ningún servidor me respondió: FAIL
- aleatoriamente, de vez en cuando, tomo uno al azar para volver
a calcular su tiempo de respuesta