

Tecnológico de Costa Rica
Escuela de Ingeniería en Computación
IC: 7602-Redes - 2 Semestre 2022
2018093728 - Paula Mariana Bustos Vargas

Prueba Corta 5-6 - V 21-10-2022

1. ¿Porqué no es posible que cada host en Internet ejecute el algoritmo de Dijkstra para encontrar la ruta mas corta hacia cualquier host en Internet? Explique (20 pts)

Si bien el algoritmo de Dijkstra se utiliza para encontrar la ruta más corta a los demás enrutadores, este necesita conocer todo el grafo de antemano como mínimo saber el peso / etiqueta del router. Note que ya para poder ejecutar de manera sencilla, es decir de un punto específico a otro necesita un alto poder de computación y memoria. Además de recordar que si se tiene que validar la eficiencia del ancho de banda y el distanciamiento entre los routers (no saltos mayores a 30)

Se presenta el caso de **host en Internet ejecute el algoritmo de Dijkstra para encontrar la ruta más corta hacia cualquier host de Internet** implicaría que se deba de conocer de antemano todo el grafo de internet, para hacerlo a una escala tan superior como es todo el internet se necesita un mayor poder computacional y de memoria, que el explicado anteriormente, además de que seria un uso ineficiente de los recursos que se posee porque puede llegar a que no se logre completar el grafo o bien tener secciones que están cambiando debido a que en ciertas circunstancias como que un router presente una congestión o se encuentre caído o inactivo esa sección del grafo, no se podría utilizar.

Además de que cabe destacar que la red del internet esta en constante cambio y crecimiento puede hasta el punto en que ya no es factible tener de antemano el conocimiento del grafo.

Por lo que la ruta también puede variar en el transcurso debido a lo mencionado anteriormente, que se caiga o tenga mucha congestión algún router que estuviera en la ruta original por lo que no seria muy viable utilizar dicho algoritmo ya que no llegaría a su destino o bien se debería de volver a calcular la ruta.

2. Explique el concepto de Hierarchical Routing (20 pts)

Es el enrutamiento jerárquico y este puede utilizar cuando la red puede crecer y por consecuencia también las tablas de enrutamiento del enrutador hasta el punto en que ya no es factible, debido a que consume memoria y más tiempo del CPU para examinarla y más ancho de banda entre los envíos de los enrutadores.

El modelo jerárquico consiste en sí que los enrutadores se dividen en regiones, donde cada enrutador conoce todos los detalles para enrutar paquetes a destinos dentro de su propia región, pero no sabe nada de la estructura interna de las otras regiones.

En la conexión de diferentes redes, cada una se considera como una región independiente, con el fin de liberar a los enrutadores de una red de la necesidad de conocer la estructura topológica de las demás.

En las redes de gran tamaño, una jerarquía de dos niveles puede ser insuficiente, por lo que sería necesario realizar multinivel donde las agrupaciones internas se subdividen aún más llegando a pasar de:

- > las regiones se agrupan en clústeres
- > los clústeres se agrupan en zonas
- > las zonas se agrupan en grupos
- > Así sucesivamente hasta que se nos agoten los nombres para clasificarlos.

Por lo que es clave determinar el número óptimo de niveles; para una subred de N enrutadores es de **$\ln N$** , y se requieren un total de **$e \ln N$** entradas por enrutador.

Como se menciona anteriormente lo que genera el enrutamiento jerárquico es que se reduce el tamaño de las tablas de enrutamiento y a medida que crece la razón entre la cantidad de regiones/subsecciones y el número de enrutadores por este, aumenta los ahorros de espacio y una mejor eficiencia de los recursos. Pero la longitud de ruta es mayor, aunque por lo general es aceptable.

3. Si tiene la siguiente subred 10.0.0.0/8 y ocupa crear 20 subredes de mínimo 160 hosts, ¿Cual máscara utilizaría? Explique en detalle. (20 pts)

Al desear 20 subredes y que cada una tenga como mínimo 160 host, se necesita de una máscara que permita tener acceso al menos de 3200 host.

La máscara que presenta la subred 10.0.0.0/8 es la 255.0.0.0 ya que está dada por /8 que si se ve en binario seria 11111111.00000000.00000000.00000000

Y el complemento de la máscara seria 0.255.255.255 el cual es la cantidad de host posibles a utilizar, que en binario se vería 00000000.11111111.11111111.11111111 que en representa 16,777,214 host posibles. Si se reparten entre 20 subredes cada una tendría aproximadamente 838860 host.

Ahora bien, si se quisiera trabajar con unas subredes que tengan menos host y más cercana al mínimo de 160 host solicitados por subred, se debería de cambiar la máscara por una superior que permita tener más de 3200 host posibles. Por lo que se utilizaría una máscara /20 que es 255.255.240.0 que en binario seria 11111111.11111111.11110000.00000000, por lo que su complemento se vería como 00000000.00000000.00001111.11111111, seria 0.0.15.255 que representa 4,094 host posibles y si se reparten entre 20 subredes cada una tendría aproximadamente 204 host.

4. El mecanismo de inundación es utilizado para distribuir paquetes de findings/updates (hallazgos) con los vecinos, comente acerca de los mecanismos para evitar que los paquetes se queden por siempre en la red. (20 pts)

Una de estas medidas es integrar un contador de saltos en el encabezado de cada paquete, que disminuya con cada salto, y el paquete se descarte cuando el contador llegue a cero

Otro mecanismo es con diques a la inundación el cual es llevar un registro de los paquetes difundidos, para evitar enviarlos una segunda vez. - Una manera seria que el enrutador de origen ponga un número de secuencia en cada paquete que recibe de sus hosts, y cada enrutador tiene una lista por cada enrutador de origen que indique los números de secuencia originados en ese enrutador que ya ha visto, si lo ha visto no lo difunde. - Ese registro seria una lista que debe de tener un contador que indique que todos los números de secuencia hasta # que ya han sido vistos. Lo que permite comprobar a la hora de llegada de un paquete si este es un duplicado y descartarlo. Nota para los elementos inferiores del contador no es necesario que se encuentren todos en la lista.

Otra manera es que los enrutadores no envían cada paquete de entrada por todas las líneas, sino sólo por aquellas que van aproximadamente en la dirección correcta e irían acompañados de alguna de los mecanismos anteriores. El cual es conocido como una inundación selectiva.

5. Para el IP 10.0.39.25/27, calcule los siguientes parametros: (20 pts)

Si /24 -> 255.255.255.0 -> 11111111.11111111.11111111.00000000

entonces:

/25 -> 11111111.11111111.11111111.10000000 -> 255.255.255.128

/26 -> 11111111.11111111.11111111.11000000 -> 255.255.255.192

/27 -> 11111111.11111111.11111111.11100000 -> 255.255.255.224

Entonces **10.0.39.25/27**

	#		Binario
#Red:	10. 0. 39. 25	----->	00001010.00000000.00100111.00011001
Mascara:	255.255.255.224	<- /27 ->	11111111.11111111.11111111.11100000
AND	-----	-----	-----
--	10. 0. 39. 0	<-----	00001010.00000000.00100111.00000000
Complemento Mascara	0. 0. 0. 31	<-----	00000000.00000000.00000000.00011111
OR	-----	-----	-----
Broadcast Address	10. 0. 39. 31	<-----	00001010.00000000.00100111.00011111

Complemento 10. 0. 39. 0 -> 00001010.00000000.00100111.00000000

- Máscara de subred (ambas notaciones)
 - Binaria: 11111111.11111111.11111111.11100000
 - 255.255.255.224
- Número de red
 - 10.0.39.25
- Broadcast Address
 - 10.0.39.31
- Lista de hosts de la red
 - 10.0.39.32
 - 10.0.39.33
 - 10.0.39.34
 - 10.0.39.35
 - 10.0.39.36
 - 10.0.39.37
 - 10.0.39.38
 - 10.0.39.39
 - 10.0.39.40

- 10.0.39.41
- 10.0.39.42
- 10.0.39.43
- 10.0.39.44
- 10.0.39.45
- 10.0.39.46
- 10.0.39.47
- 10.0.39.48
- 10.0.39.49
- 10.0.39.50
- 10.0.39.51
- 10.0.39.52
- 10.0.39.53
- 10.0.39.54
- 10.0.39.55
- 10.0.39.56
- 10.0.39.57
- 10.0.39.58
- 10.0.39.59
- 10.0.39.60
- 10.0.39.61