

Yönlendirme Algoritmaları Nedir?

Çalışma Kağıdı

Yönlendirme algoritmaları, ağdaki cihazlar arasındaki en iyi yolu belirleyerek veri paketlerinin hedefe ulaşmasını sağlayan protokollerdir. Ağ topolojisi, bant genişliği ve gecikme gibi faktörleri göz önünde bulundurlar.

Sorular

1. Aşağıdakilerden hangisi bir Durum (Link State) yönlendirme algoritmasıdır?

- A) RIP
- B) BGP
- C) OSPF
- D) IGMP

2. Bağışıklık (Distance Vector) algoritmalarının temel dezavantajı nedir?

- A) Yüksek bant genişliği tüketimi
- B) Hesaplama karmaşıklığı
- C) Döngü (loop) oluşumu riski ve yavaş yakınsama
- D) Tüm ağ topolojisini bilme gerekliliği

3. Yönlendirme algoritmalarının temel amacı nedir?

- A) Ağ güvenliğini sağlamak
- B) Veri paketlerinin hedefe en iyi yoldan iletilmesini sağlamak
- C) Ağdaki cihaz kimliklerini doğrulamak
- D) Ağ trafiğini izlemek

4. Dijkstra Algoritması (SPF) Örneği

5. Bellman-Ford Algoritması (Distant Vector) Örneği

6. Tanımla: Yönlendirme Algoritması Nedir?

7. Tanımla: Bağışıklık Algoritmaları (Distance Vector) Nedir?

8. Tanımla: Durum Algoritmaları (Link State) Nedir?

Cevap Anahtarı

1. C) OSPF — OSPF (Open Shortest Path First), bir Durum (Link State) yönlendirme protokolüdür. Ağdaki tüm yönlendiricilerin ağ topolojisinin tam bir haritasını tutmasını sağlar.
2. C) Döngü (loop) oluşumu riski ve yavaş yakınsama — Bağımsızlık algoritmaları, yönlendirme döngüleri oluşturmaya yatkındır ve ağdaki değişikliklere yavaş tepki verebilirler (yavaş yakınsama).
3. B) Veri paketlerinin hedefe en iyi yoldan iletilmesini sağlamak — Yönlendirme algoritmalarının birincil görevi, veri paketlerinin bir kaynaktan hedefe en verimli ve hızlı şekilde ulaşmasını sağlayacak yolu belirlemektir.
4. 1. Başlangıç düğümünü seçin ve diğer tüm düğümlere uzaklığını sonsuz olarak atayın. 2. Başlangıç düğümünün uzaklığını 0 olarak ayarlayın. 3. Ziyaret edilmemiş düğümler kümesini oluşturun. 4. Mevcut düğümden komşu düğümlere olan mesafeleri güncelleyin. 5. En kısa mesafeye sahip ziyaret edilmemiş düğümü seçin ve ziyaret edildi olarak işaretleyin. 6. Tüm düğümler ziyaret edilene kadar 4. ve 5. adımları tekrarlayın.
5. 1. Her düğüm, kendi uzaklık vektörünü (hedef düğümlere olan mesafeler) başlangıçta sonsuz olarak ayarlar, sadece kendine mesafesi 0'dır. 2. Her yönlendirici, komşularından gelen uzaklık vektörlerini alır. 3. Her yönlendirici, aldığı komşu vektörlerini kullanarak kendi uzaklık vektörünü günceller: Kendi mesafesi + komşunun mesafesi, eğer daha kısaysa güncellenir. 4. Bu işlem, ağdaki mesafeler sabitlenene kadar (genellikle V-1 iterasyon) tekrarlanır, burada V düğüm sayısıdır.
6. Ağdaki veri paketlerinin hedefe ulaşması için en iyi yolu belirleyen algoritma.
7. Komşularından aldığı bilgilere göre kendi yönlendirme tablosunu oluşturan algoritmalar (örn: RIP).
8. Ağın tamamının topolojisini bilen ve her yönlendiricinin tam haritasını çıkaran algoritmalar (örn: OSPF).

Bounlu

Tüm kartlar, adım adım çözümler ve AI hoca desteği Notek uygulamasında.
Sınav tarihlerini Promy otomatik hatırlatıcıya çevirir.