

Име: Алекс Узунов

Специалност: Софтуерно инженерство

Факултетен номер: 7MI0600096

I. Сравнете динамичните маршрутизиращи протоколи: **“distance vector”** и **“link-state”**.

- **Обмен на информация:** При **“distance vector”** цялата таблица с маршрути се разпространява към всички съседни рутери в мрежата, докато при **“link-state”** се разпространява само информация за състоянието на връзките между рутерите.
- **Обновяване на информацията:** **“distance vector”** обновява информацията на всеки определен интервал или при промяна в мрежата, докато **“link-state”** обновява информацията само при промяна във връзките.
- **Алгоритъм за вземане на решение:** **“distance vector”** използва прост алгоритъм за избиране на най-добрия маршрут въз основа на разстоянието до дестинацията, докато **“link-state”** използва по-сложни алгоритми като алгоритъма на Дейкстра за намиране на най-кратък път върху графа на базата на информацията за състоянието на връзките.
- **Сходимост:** **“distance vector”** изисква повече време за постигане на сходимост в мрежата поради итеративния обмен на информация със съседните рутери. Могат да се появят временни неправилности в мрежата преди да се постигне стабилно състояние. **“link-state”** има по-бърза сходимост, тъй като всяка промяна във връзките се изпраща на всички рутери, което позволява по-бързо и по-стабилно настройване на мрежата.

- **Мащабируемост:** **“distance vector”** не е толкова мащабируем както **“link-state”**, особено в големи мрежи с много рутери. Обемът на информацията, която се разпространява, нараства с размера на мрежата. **“link-state”** е по-мащабируем от **“distance vector”** протоколите. Информацията за състоянието на връзките се предава само при промяна, което позволява по-ефективно използване на ресурсите и го прави подходящ за големи мрежи.
- **Откриване на проблеми:** **“distance vector”** не е толкова добър в откриването на проблеми в мрежата, тъй като информацията се разпространява постепенно и не се знае моменталното състояние на всички връзки, докато **“link-state”** протоколите са по-добри в откриването и реагирането на проблеми в мрежата, позволявайки бързо преориентиране на трафика в случай на отказ или промяна в топологията.
- **Разходи:** **“distance vector”** протоколите са по-евтини за имплементация и поддръжка, тъй като изискват по-малко обработка и по-малко използване на ресурси, докато **“link-state”** протоколите са по-скъпи за имплементация и поддръжка, тъй като изискват по-сложни алгоритми и по-голям обем на разпространение на информация.

II. Опишете стъпките за процеса на договаряне при транспортния протокол TCP.

Процесът на договаряне при транспортния протокол TCP включва следните стъпки:

- 1. Изпращане на заявка за връзка (SYN) -**
клиентът установява връзката като изпраща TCP сегмент със знаменце SYN (Synchronize) и избира произволен източников порт. В сегмента се включват синхронизационни номера (sequence numbers), за да се синхронизират последователните байтове между клиента и сървъра.
- 2. Потвърждение на заявката за връзка (SYN-ACK)**
- Сървърът получава TCP сегмента със знаменце SYN и създава ново TCP съединение. Сървърът изпраща отговорен TCP сегмент, който включва знаменце SYN-ACK (Synchronize-Acknowledgment) и потвърждава заявката за връзка от клиента. В сегмента се включват и синхронизационни номера, които потвърждават синхронизацията между двата крайни устройства.
- 3. Потвърждение на потвърждението (ACK) -**
Клиентът получава TCP сегмента със знаменце SYN-ACK от сървъра и изпраща потвърждение (ACK) за потвърждението на заявката за връзка. В сегмента се включва информация за потвърждаване на номерата на сегментите, които са получени.

След завършване на горните три стъпки, процесът на договаряне при TCP е приключил и връзката е установена. От този момент клиентът и сървърът могат да започнат да изпращат данни в рамките на това TCP съединение.

III. Посочете транспортния протокол и портовете по подразбиране (well-known ports) на следните услуги:

- SMTP

Транспортен протокол: TCP

Портове по подразбиране: 25 (за имейл сървъри), 587 (за имейл клиенти)

- SNMP

Транспортен протокол: UDP

Портове по подразбиране: 161, 162

- DNS

Транспортен протокол: UDP

Портове по подразбиране: 53

- DHCP

Транспортен протокол: UDP

Портове по подразбиране: 67 (за сървъри), 68 (за клиенти)

- HTTP

Транспортен протокол: TCP

Портове по подразбиране: 80

- HTTPS

Транспортен протокол: TCP

Портове по подразбиране: 443

- FTP

Транспортен протокол: TCP

Портове по подразбиране: 21

- SSH

Транспортен протокол: TCP

Портове по подразбиране: 22

- POP3

Транспортен протокол: TCP

Портове по подразбиране: 110

- IMAP

Транспортен протокол: TCP

Портове по подразбиране: 143

- TELNET

Транспортен протокол: TCP

Портове по подразбиране: 23