



استاد: دکتر هادی برقی

درس: شبکه های پیشرفته

## Preamble:

بیشتر تمرکز روی امتحان پایانی است. نمره‌ی درس از بین ۲۲ تا ۲۳ نمره است.

۱۰ درصد نمره حل تمارین است

۱۰ درصد نمره ترجمه مقاله (به صورت نمره اضافی و ارفاقی)

۵ درصد کوییز کلاسی است که احتمالاً حذف گردد.

۷۵ درصد امتحان پایانی است.

درس کاربردی است و امتحان از اسلاید هاست. بیشتر امتحان نهایی تستی است و فقط چند تا تشریحی دارد.

انتخاب و تایید مقاله با خودماس است. مقاله نباید مروری باشد . لپ کلام مقاله در کمتر از یک صفحه باید آورده شود (مشکل - نتایج- راه حل) . موضوع باید مرتبط با درس باشد و حتما انگلیسی باشد Impact Factor مقاله باید بین  $Q^1$  تا حداقل  $Q^3$  باشد.

– انتهای Introduction و قسمت پیشنهادات در بخش چهارم مقاله مهم هستند.

نوشتن نقد مقاله و خوبی و بدی هاش مثلا ایراد ها و نقصی های مقاله.

اصل مقاله و کار ما باید تا انتهای ترم در Lms بارگذاری شود. یکسان نبودن مقاله هم بهتر است از روی گروه چک شود.

سایت های پیدا کردن مقاله:

[Scholar.Google.Com](#)

[Ieeexplore.Com](#)

سایت تایین رتبه بندی مقالات :

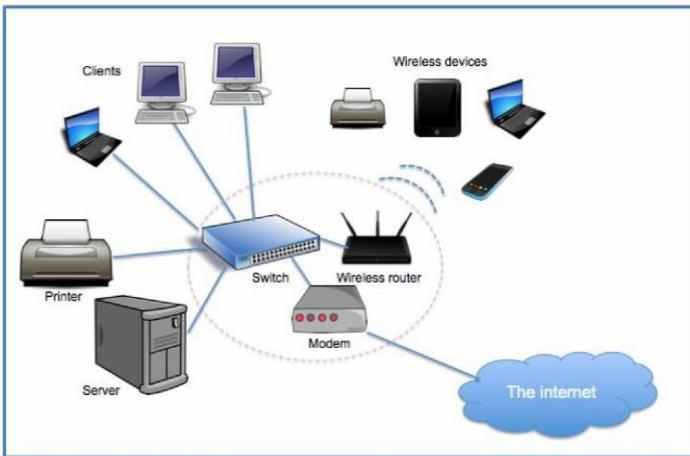
[Scimagojr.Com](#)

## جلسه دوم:

### شبکه کامپیوتری چیست؟

- دو یا بیش از دو کامپیوتر یا ابزارهای جانبی متصل به هم
- تجهیزات سخت افزاری و نرم افزای = منبع
- دلایل استفاده از شبکه:

- اشتراک منابع
- کاهش هزینه
- قابلیت اطمینان
- کاهش زمان
- قابلیت توسعه
- ارتباطات
- تسهیل امور



12 of 19

تعريف شبکه: وقتی چند کامپیوتر یا دیوایس دیگر را به هم متصل میکنیم و تبادل داده میکنند.

اتصال ماوس به کامپیوتر شبکه نیست

Resource: تجهیزات سخت افزاری و نرم افزاری که در فرایند ارتباط مورد استفاده قرار

میگیرند. مثال: کارت شبکه - کابل - نرم افزار شبکه

## شبکه کامپیوتری چیست؟

- موارد مورد نظر در طراحی شبکه
  - اندازه سازمان
  - سطح امنیت
  - نوع فعالیت
  - سطح مدیریت
  - مقدار ترافیک
  - بودجه
- یک کامپیوتر در شبکه = یک ایستگاه کاری، یک گره

یک کامپیوتر در شبکه میتواند یک ترمینال نود یا گره انتهایی باشد.

اما فایر وال گره انتهایی نیست

## مدل های شبکه

- یک کامپیوتر : هم سرویس دهنده، هم سرویس گیرنده
- سرویس دهنده : گره ارائه دهنده یک سرویس (فایل، محاسبات، ترجمه آدرس و ...)
- سرویس گیرنده : گره درخواست کننده یک سرویس

- انواع مدل های شبکه:
  - سرویس گیرنده / سرویس دهنده
  - نظیر به نظیر

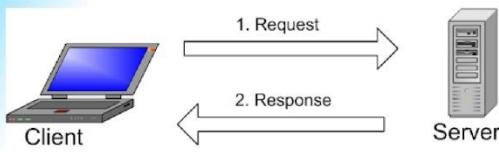
انواع شبکه :

Client/Server

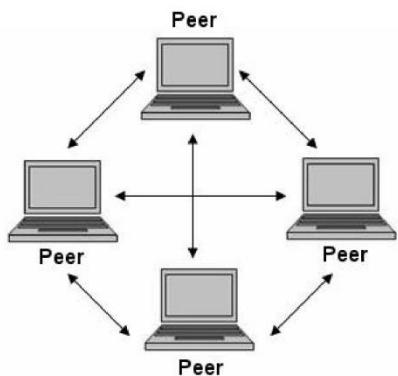
Peer To Peer

آدرس سروس همیشه باید توسط Dns در دسترس باشد

## مدل های شبکه



- سرویس گیرنده / سرویس دهنده
- یک سرویس دهنده همیشه روش درخواست سرویس : سرویس گیرنده
- ارائه سرویس : سرویس دهنده



- نظیر به نظیر
- سرویس دهنده همیشه روش نیاز نیست
- ارتباط بین سرویس گیرنده ها

15 of 19

در کلاینت سروری شروع درخواست ها با کلاینت است اما در نظیر به نظیر هر نودی میتواند شروع کننده ارتباط باشد

در شبکه نظیر به نظیر انتقال اطلاعات از لحاظ زمان بهینه تر است همچنین یک نود میتواند همزمان با چندین نود ارتباط برقرار کند

## اجزای شبکه

### ۰ گره ها



### ۰ رسانه



### ۰ واسط شبکه



16 of 19

گره : میتواند نود انتهایی یا نود میانی باشد . یو اس بی نود انتهایی محسوب نمیشود و کامپیوتر یک نود انتهایی است اما پرینتر تحت شبکه نود انتهایی محسوب میشود .

رسانه: کابل کواکسیال که قدیبی تر است و نویز الکترومغناطیسی کمی روی آن اثر میگذارد - زوج به هم تابیده (Twisted Pair) که استفاده از آن آسان است و نویز پذیری بیشتری دارد مدل های متنوعی دارن Cat<sup>5</sup> – Cat<sup>6</sup>-Cat<sup>7</sup> تفاوت ها هم در سرعت است به علت تفاوت در جنس ساخت و میزان تابیدگی رشته سیم ها - Optical Fiber یا فیبر نوری که نور را عبور میدهد سرعت بالایی دارد سمت چپ تصویر.

واسط شبکه Network Interface Card : داده را از نود به رسانه برای ارسال و از رسانه تشخیص و به نود برای پردازش انتقال میدهد

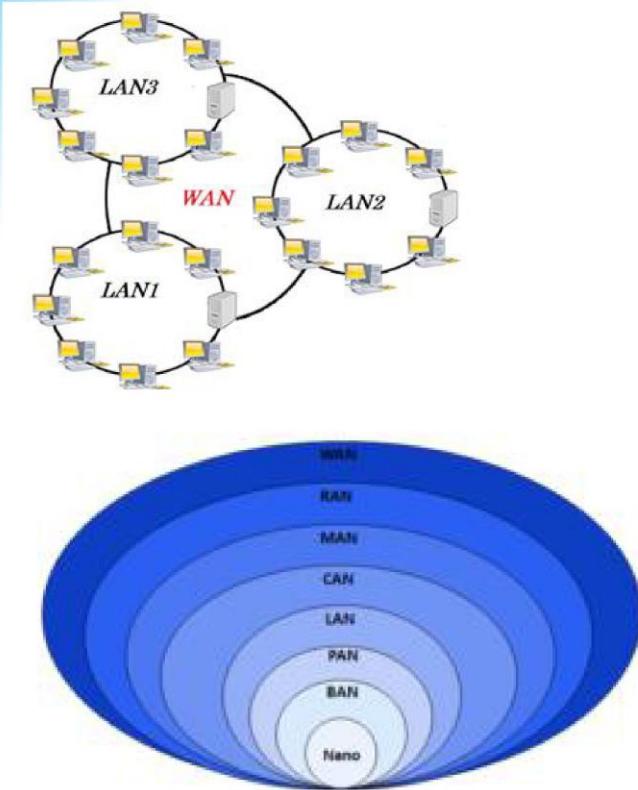
تصویر سمت راست فیبر نوری و مازول Sfp برای اتصال کابل فیبر و سمت چپ کارت زوج به هم تابیده است



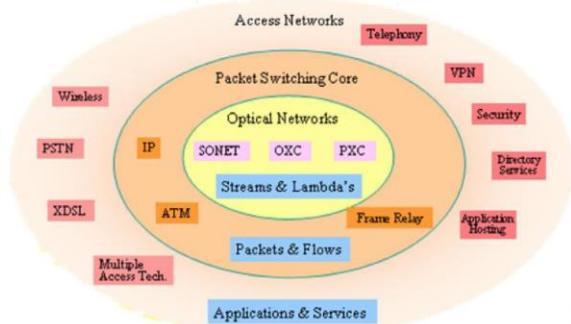
در حد یک ساختمان و در نهایت یک محوطه با ساختمان های داخل آن است نه به گستردگی یک شهر

ارتباط در اتاق سرور از نوع ارتباط زیرساخت است.

## انواع شبکه از لحاظ گستردگی



- شبکه های گستردگ (WAN)
- منطقه بزرگ و بسیار بزرگ
- سرعت کم
- تاخیر زیاد
- شامل انواع مختلف شبکه
- بزرگترین و مهم ترین: اینترنت



18 of 19

Wan: ارتباط بین چند Lan است.

در شکل سمت راست بیان شده که شبکه های بیرونی تر از شبکه های درونی برای انتقال اطلاعات استفاده میکنند

مثال: Packet Switching Core ها از Access Network است: میکنند

مثال: Access Network : دسترسی یوزر به شبکه گستردگ تر را فراهم میکند (End User) –Telephony-Vpn برنامه ها و سرویس ها در این سطح از شبکه است

Access Network با سایر شبکه ها . بسته هاو جریان Packet Switching Core واسط انتقالی در این قسمت است

Optical Network : شبکه زیر ساخت است . استریم ولاندا در این سطح از شبکه صورت میگیرد

## توپولوژی شبکه

- نحوه اتصال کامپیوترها(گره ها) به یکدیگر
- پارامترهای اصلی در طراحی : قابل اعتماد بودن، مقرون به صرفه بودن

### انواع توپولوژی:

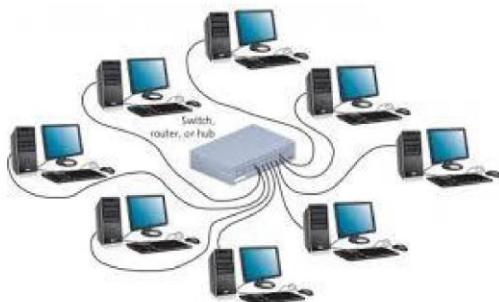
#### توپولوژی ستاره (Star)

عیب : خرابی هاب = از کار افتادن شبکه

مزایا :

- برپایی ساده
- توسعه ساده

خرابی یک گره = فقط قطع ارتباط همان گره



3 of 19

توپولوژی به معنی ریخت شناسی است . شکل ارتباطی نود ها به هم را توپولوژی گویند

قابل اعتماد بودن (Reliable) : با قطع شدن قسمتی از ارتباط کل ارتباط قطع نشود.

شبکه استار در اکثر Lan های امروزی به کار میروند.

نقش مرکزی توسط سوییچ انجام می‌شود یعنی ما Single Point Of Failure داریم و با خراب شدن سوییچ شبکه از کار می‌افتد.

## توپولوژی شبکه

### • توپولوژی حلقه (Ring)

• معایب:

- خرابی یک گره = از کار افتادن شبکه
- سخت افزار پیچیده و گران
- غیر پل‌آگ آند پلی

• مزایا:

- بروایی ساده
- توسعه ساده



4 of 19

توپولوژی Ring هنوز هم استفاده می‌شود اما قبل از بیشتری داشته

سخت افزار Network Interface Card پیچیده و گران است

نمی‌توانیم بدون قطعی شبکه نودی را کم یا زیاد کنیم

## توپولوژی شبکه

### • توپولوژی Bus

• معایب :

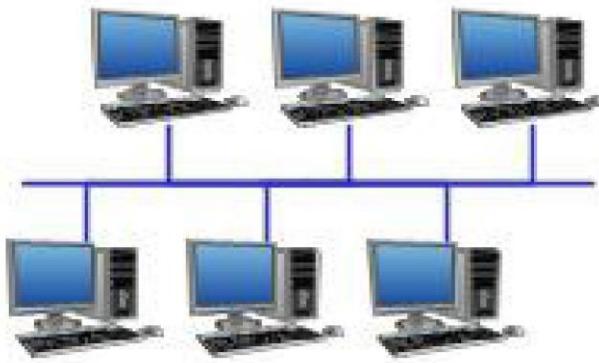
• خرابی کابل رابط اصلی = از کار افتادن شبکه

• مزایا :

• برپایی ساده

• توسعه ساده

• هزینه کم



## توپولوژی شبکه

### • توپولوژی توری (Mesh)

• معایب :

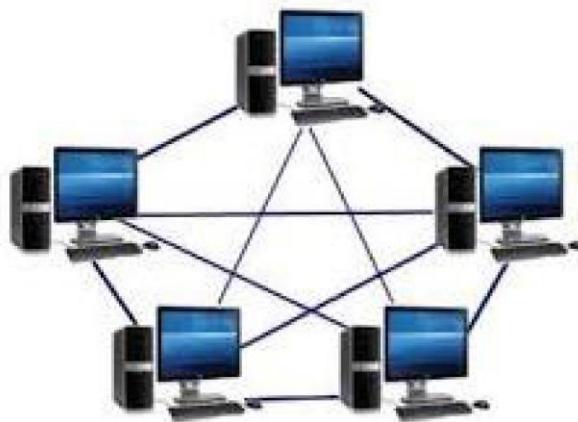
• تعداد زیاد خطوط ارتباطی

• عدم صرفه اقتصادی

• مزایا :

• درجه امنیت و اطمینان بالا

• فعالیت در صورت قطع یک یا چند ارتباط



6 of 19

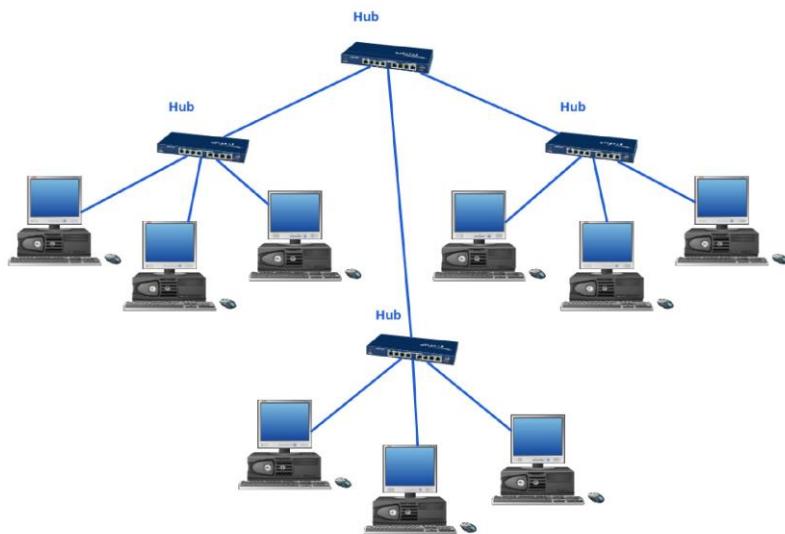
توپولوژی مش بیشتر در شبکه های بیسیم استفاده میشود

استفاده بصورت Full Mesh هزینه بسیار بیشتری دارد و خیلی قابل توسعه نیست

در فول مش هر دو نود ارتباط مستقیم با هم دارند و نود میانی بین آنها نیست و از لحاظ امنیتی بسیار امن است . همچنین با قطع شدن یک یا چند نود شبکه میتواند به کار خود ادامه دهد

## توپولوژی شبکه

- **توپولوژی درخت (Tree)**
- مشابه توپولوژی ستاره، همان مزایا و معایب

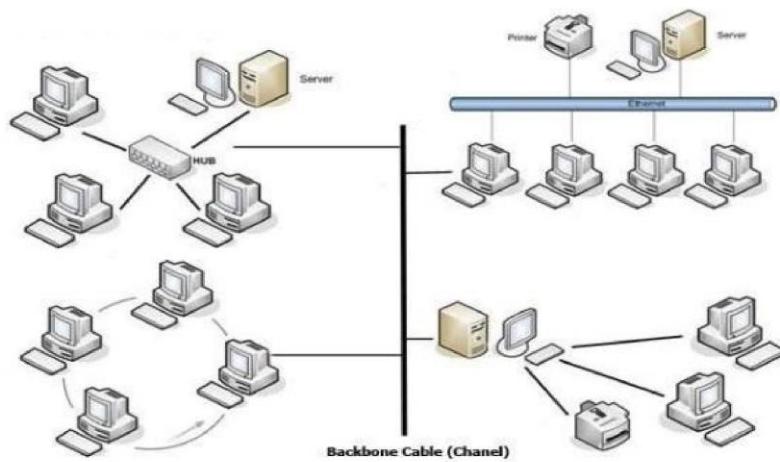


7 of 19

اگر هاب را به نود بالاتری وصل کنیم شکل درخت می‌گیرید

## توبولوژی شبکه

- توبولوژی ترکیبی (Hybrid)
- ترکیبی از چند توبولوژی
- با یک کابل اصلی (Back Bone) به هم متصل می شوند
- توسط Back Bone به Bridge وصل می شود



8 of 19

ارتباط بین شبکه های مختلف را با Back Bone یا ستون فقرات برقرار کنیم و شبکه ها از طریق Back Bone به Bridge متصل میشود.

## مدل OSI

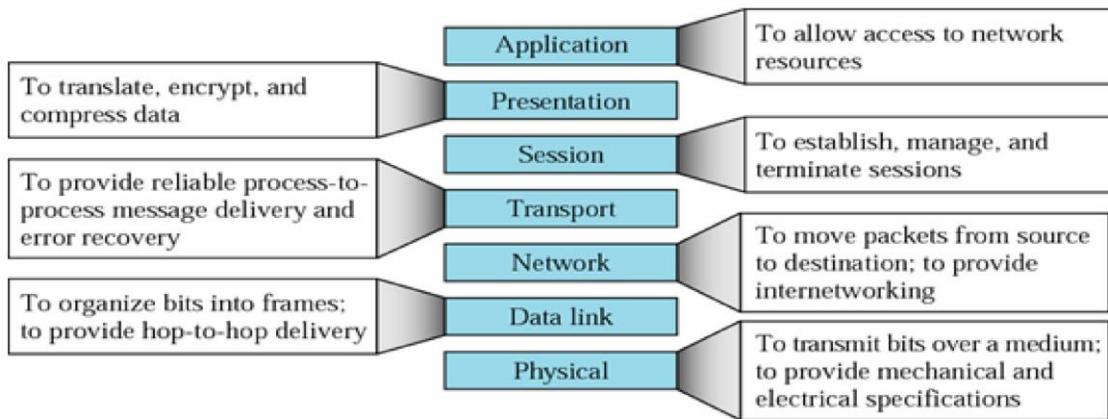
• سازمان استاندارد جهانی ISO  
Open System Interconnect •

• دارای ۷ لایه است  
• لایه ها:

- |                     |                   |
|---------------------|-------------------|
| <b>Physical</b>     | • لایه فیزیکی     |
| <b>Data link</b>    | • لایه پیوند داده |
| <b>Network</b>      | • لایه شبکه       |
| <b>Transport</b>    | • لایه انتقال     |
| <b>Session</b>      | • لایه اجلاس      |
| <b>Presentation</b> | • لایه نمایش      |
| <b>Application</b>  | • لایه کاربرد     |

لایه ها بصورت بر عکس لیست شده اند (از پایین به بالا)

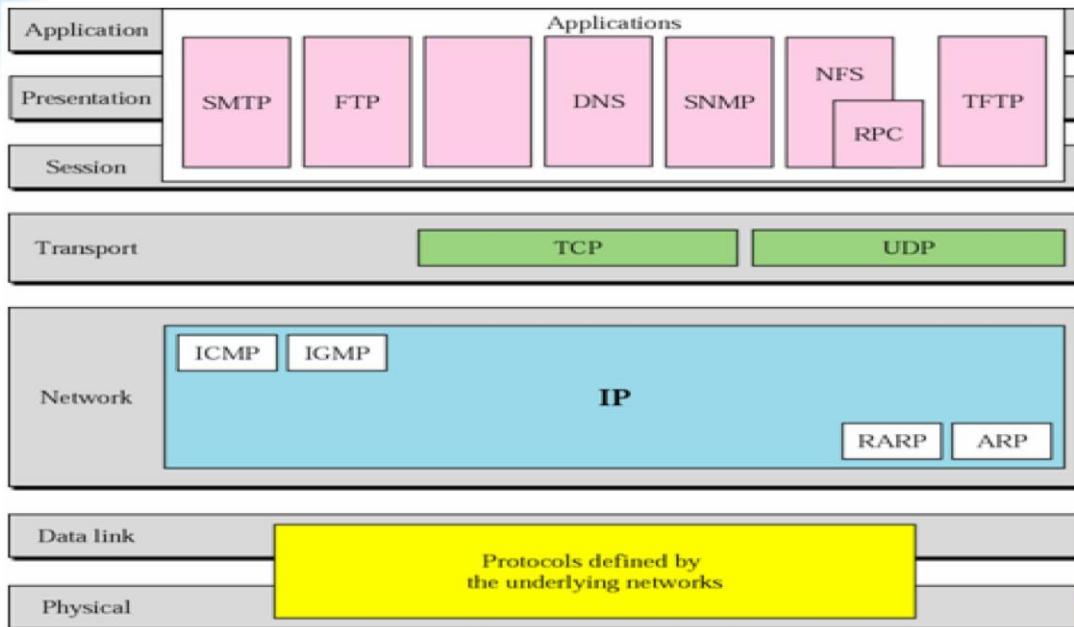
# مدل OSI



10 of 19

وظایف لایه ها قابل مشاهده است.

## مدل OSI و مدل TCP/IP



11 of 19

مدل Osi اول استاندارد شد اما هیچ وقت پیاده سازی نشد .

مدل Tcp/Ip اول پیاده سازی شد بعد مدل شد که الان در اینترنت استفاده میشود

این شکل جایگاه لایه های Osi در مدل Tcp/Ip را نشان میدهد.

Underlyering Network با هم Datalink , Physical

Application لایهی Application . Presentation . Session است

## پروتکل های شبکه

### TCP/IP •

- مهم ترین و پر کاربردترین پروتکل ها
- لایه انتقال : TCP •
- لایه شبکه : IP •

### Net Bios •

- واسط توسعه یافته توسط IBM
- مرتبط با لایه اجلاس : ایجاد ارتباط بین کامپیوترها از طریق شبکه
- در سیستم های جدید: NetBIOS over TCP/IP (NBT)

در لایه ی ان Tcp

Ip در لایه ی انتقال هست و همه ی شبکه ها آنرا می فهمند و مثل چسب شبکه ها را متصل می کند.

## ابزارهای اتصال دهنده Connectivity Devices

<b>Repeater</b>	• تکرار کننده
<b>HUB</b>	• هاب
<b>Router</b>	• مسیریاب
<b>Gateway</b>	• دروازه
<b>Bridge</b>	• پل
<b>Switch</b>	• سوئیچ

نود های میانی:

برای گسترش (Extend) دادن شبکه استفاده میشود . مثال : طول کابل زوج به هم تابیده ۱۰۰ متر میتواند باشد و برای افزایش طول در ۱۰۰ متر یا ۹۰ متر یکی قرار میدهیم

برقرار کننده ارتباط دو شبکه با پروتکل های مختلف است مثال: ارتباط بین شبکه بیسیم و سیمی که هر دو در لایه ۲ هستند (یکی ۸۰۲.۱۱ و یکی ۸۰۲.۳) کارش در اصل تبدیل پروتکل است.

یک تجهیز لایه ۲ است و بسته ها را باز میکند و میتواند آها را از روی آدرس مقصد هدایت کند . جدول هم دارد . خیلی از کار های روترا انجام میدهد

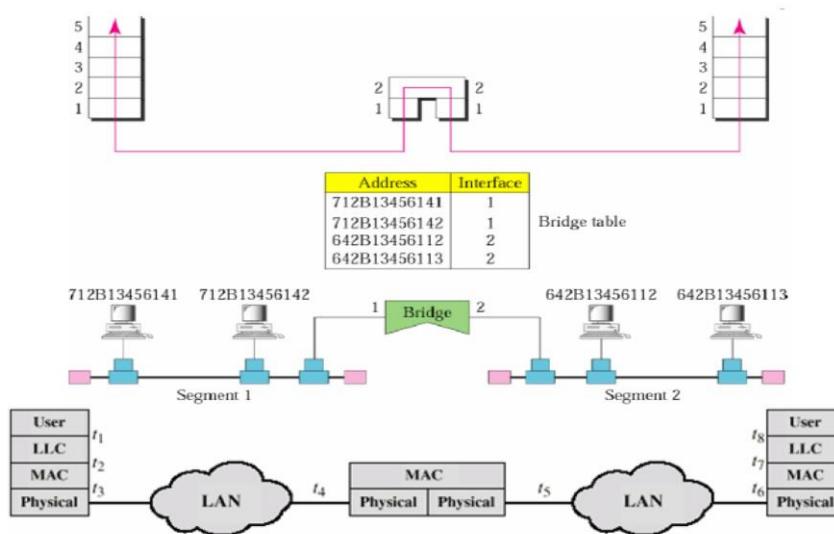
نقطه ارتباطی بین یک شبکه و بقیه ای شبکه هاست مثال : در یک شبکه واي فای اکسس پویت که به اینترنت وصل است Gate Way (در واژه) است.

کار مسیر یابی را بر اساس آدرس مقصد انجام میدهد Router

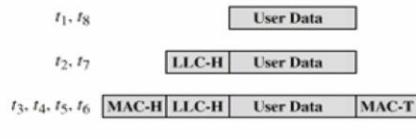
یک ابزار پسیو است و فقط هر پکتی که دریافت میکنه به صورت کور کورانه روی بقیه پورت ها ارسال میکند. فقط ارتباط برقرار میکند مثل سیم. با ارسال همزمان دو نود تداخل دارد اما سوییچ و روترا تداخل ندارند. مثل شبکه Bus . یک تجهیز لایه یک است .

# ابزارهای اتصال دهنده Connectivity Devices

پل



(a) Architecture



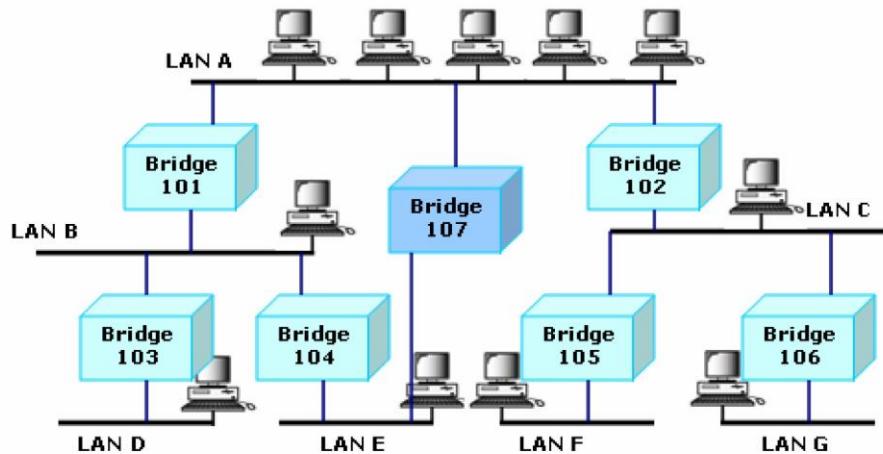
(b) Operation

14 of 19

در شکل کار برد پل و هم تفاوت پروتکل ها مشهود است . تا لایه ۲ بسته ها را باز میکند.

# ابزارهای اتصال دهنده Connectivity Devices

## ۰ پل - دستیابی چندگانه



## ۰ درخت پوشایشی

15 of 19

هر کدام از این Lan ها میتوانند پروتکل های متفاوتی داشته باشند . کار بریج ترجمه‌ی پروتکل است.

تشخیص اینکه مقصد کدام شبکه است کار پل از . ممکن از در این فروارد کردن لوبوپ یا حلقه رخددهد برای جلو گیری از آن بریج یک درخت پوشایشی ایجاد میکند . این درخت شامل تمام نود ها میشود و حلقه ندارد . روت یا گره ریشه در درخت پوشایشی است که آیی دی بزرگتر را دارد .

۱۰۷ بریج روت ما هست و بریج ۱۰۱ و ۱۰۲ زیر شاخه ۱۰۷ هستند

۱۰۶ و ۱۰۵ زیر شاخه ۱۰۲

۱۰۴ و ۱۰۳ نیز زیر شاخه ۱۰۱ هستند

در این حالت ارتباط ۱۰۷ با Lan E باید قطع شود تا ما درخت داشته باشیم و این کار بصورت Logically اتفاق می‌افتد.

یا اینکه در حالت دیگر ارتباط ۱۰۴ با Lan B باید قطع شود تا درخت تشکیل شود و لوب نداشته باشیم

## ابزارهای اتصال دهنده Connectivity Devices

### سوییچ

#### Cut-through

ارسال بر اساس چهار بایت اول (آدرس مقصد) همزمان با دریافت

#### Store-and-forward

ارسال پس از دریافت کامل بسته

سوییچ‌ها برای فروارد کردن از دو متد استفاده می‌کنند بر این اساس دو نوع سوییچ داریم

در ابتدا همه‌ی بسته را در یافتن می‌کنند سپس تصمیم می‌گیرد به کجا ارسال کند. برای حالتی است که نیاز به پردازش در سمت سرورداریم کاربرد دارد.

در Cut-Through سوییچ به پروتکل لایه‌ی بالا یا IP اصلاحکاری ندارد

پس از دریافت ۶ بایت اول بسته را ارسال می‌کند که تاخیر در آن کمتر است.-(اشتباه نوشته ۴ بایت)

## نسل های مختلف LAN

• نسل اول : CSMA/CD

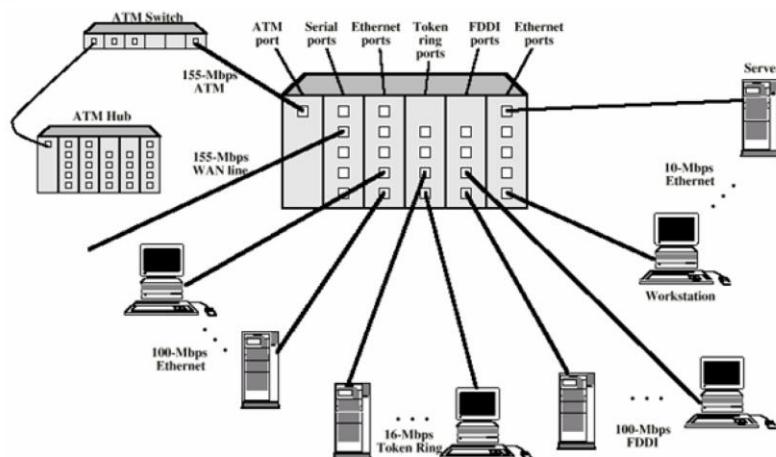
• نسل دوم : Fiber Distributed Data ) FDDI (Interface

• نسل سوم : ATM LAN

• هاب چند پروتکلی

• فرخ بالای ATM

• شبکه های رو به رشد



17 of 19

نسل اول : قابلیت Collision Detection داشتیم و در پروتکل های Bus – Ring –Hub یک محدوده Collision داریم مثال: اگر ۱۰ نод به یه توپولوژی بس متصل باشند تشکیل یک Contention Domain یا محدوده رقابت دارند و اگر با همزمان ارسال کنند Collision رخ میدهد و نسل اول قابلیت تشخیص این خطا را با مقایسه داده ارسالی و در یافتنی را داشت که پس از تشخیص ارسال متوقف میشود و یک وقفه زمانی رندوم ایجاد میشود و داده مجدد ارسال میگردد.

نسل دوم : Lan - Fddi معرفی فیبر نوری بود و مديا نیز فیبر نوری بود که توپولوژی کمتر استار است و توپولوژی قالب رینگ هست.

نسل سوم : Multi-Protocol است و در این نسل برای حال و آینده میتواند شبکه مبتنی بر Ip و مبتنی بر Atm می توانند به راحتی کنار هم کار کنند.

## ارسال سیگنال

- پهنهای باند : محدوده فرکانسی
- معادل میزان داده ارسالی در واحد زمان
- بر حسب بیت بر ثانیه
- روش های ارسال:
  - باند پایه **baseband**
  - پهن باند **broadband**
- در شبکه محلی سیمی : باند پایه، بدون تقسیم پهنهای باند
- داده ها ← بسته ها
- ارتباط
  - نیمه دو طرفه **half duplex**
  - کاملا دو طرفه **full duplex**

پهنهای باند : محدوده ای فرکانسی که داده در آن محدوده جابجا میشود گویند. مثال: فرستنده ی بسیسم داده را روی سیگنال آنالوگ سوار میکند تا بتواند ارسال کند و این محدوده پهنهای باند است و با داده ارسالی Data Rate رابطه دارد Bitp/S

داده ها در قالب بسته اند و استریمی نیستند.

در Full Duplex همزمان هر دو نمیتوانند ارسال و در یافت کنند اما در Half Duplex دو میتوانند همزمان ارسال و دریافت کنند

## جلسه ی سوم:

### رسانه

- هدایت شده : سیم و فیبر
- حرکت سیگنال در مسیری مشخص
- هدایت نشده : بیسیم
- حرکت سیگنال به همه طرف و بدون مسیر تعیین شده

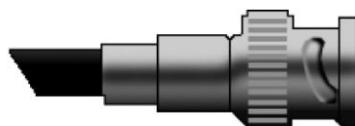
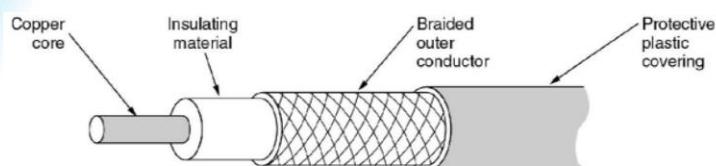
مدیا یا رسانه: رسانه چیزی هست که سیگنال یا دیتا داخل اون از مبدأ به سمت مقصد هدایت میشود.

رسانه هدایت نشده = هوا

### رسانه های هدایت شده

- کابل شبکه، وابسته به توپولوژی

#### • کابل هم محور coaxial



#### • Thick و Thin

#### • اتصال BNC

#### • توپولوژی bus و ring

• ۱۰ مگابیت در ثانیه : 10Base2 ، 10Base5 •

تفاوت نوع Thick و Thin در قطر هسته است.

نام کانکتور آن Bnc است

## رسانه های هدایت شده

### • UTP : زوج سیم تاییده بدون حفاظ

- ارزان
- نصب آسان
- سبک و منعطف
- نرخ : تا **100Mbps**
- مسافت : **100m**

- حساس به نویز

### • اتصال RJ45

- دسته های مختلف : **CAT7** تا **CAT1**
-  **CAT3**
-  **CAT5**



• دونوع Utp و Stp : Twisted Pair

سرعت کابل های کنونی خیلی بیشتر از ۱۰۰ مگابایت بر ثانیه نیز هست

نسبت به Coaxial به نویز حساس تر است

تفاوت Cat ها در جنس سیم و تاییدگی است .

## رسانه های هدایت شده

• **STP** : زوج سیم تابیده دارای حفاظ

• نویز پذیر کمتر

• گران تر

• مسافت: 500m

• اترنت سریع

**100Mbps**, انتقال روی ۲ زوج، سرعت: UTP CAT5 : 100BaseTX •

**100Mbps**, انتقال روی ۴ زوج، سرعت: UTP CAT5 : 100BaseT4 •

یک گیگ بر ثانیه را هم سائورت میکند

## رسانه های هدایت شده

• **فیبر نوری**

• از جنس شیشه

• سیگنال: پالس نوری

• مقاوم در برابر نویز

• فاصله زیاد: بیش از 100km

• **أنواع فiber:**

• تک حالت (single mode) : قطر فیبر بسیار کم: 8.3um، سیگنال: لیزر  
تک پرتو، گران، غیر منعطف

• چندحالت (multi mode) : قطر فیبر زیاد: 62.5um، سیگنال: نور LED  
چند پرتو، ارزان تر از تک حالت، منعطف، مسافت کمتر از تک حالت

صد در صد در برابر نویز الکترو مغناطیسی مقاوم است

در حالت تک حالت خمیدگی باعث افت شدید پرتو های نوری میشه و شاید خراب بشه ۱۰۰ کیلومتر و بیشتر ارسال میکند حتی تا ۲۰۰ کیلومتر

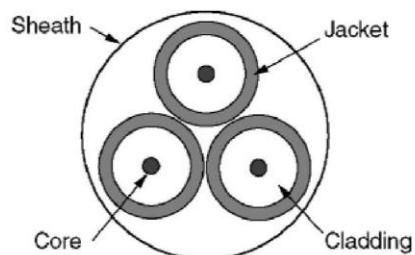
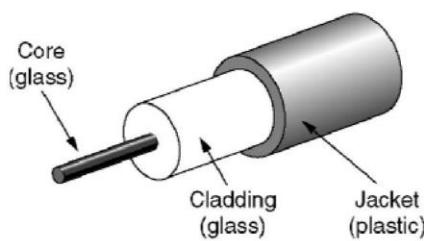
چند حالته مسافت زیر ۱۰۰۰ متر مورد استفاده است

## رسانه های هدایت شده

### • فیبر نوری

100Mbps : سرعت : 100BaseFX •

• سرعت ها بسیار بالاتر : 40Gbps , 25Gbps , 10Gbps , 1Gbps  
100Gbps



جدیدا ۲۰۰ گیگ ۴۰۰ گیگ دارد و ۸۰۰ گیگ هم تست شده .

### رسانه هدایت نشده:

تنها رسانه هدایت نشده هوا هست و سیگنال بیسیم در هوا پخش میشود و توسط گیرنده آنتن دریافت میشود

## کارت شبکه NIC

- واسطه بین کامپیوتر و شبکه
- اتصال از طریق یکی از باس ها : **PCI**، **ISA**، **USB** یا **on board**
- به همراه درایور : لایه های فیزیکی و پیوند داده

### • وظایف :

- قاب بندی داده لایه شبکه : عملیات لایه پیوند داده
- کدگزاری و سایر عملیات لایه فیزیکی
- ارسال و دریافت داده

دیتای داخل کامپیوتر را روی کابل قرار میدهد و بلعکس.

قدیمی تر است . Isa

IP  
ARP  
RARP

## آدرس دهی در شبکه های کامپیوتري

### • انواع آدرس

- آدرس فیزیکی  
**MAC Address** یا **Physical Address** .

- آدرس IP
- آدرس لایه شبکه
- شماره درگاه
- آدرس لایه انتقال

داده ها در قالب بسته ها از مبدا به مقصد هدایت می شودند.

برای هدایت درست مقصد باید آدرس درستی داشته باشد.

آدرس فیزیکی یا Mac در لایه ی Datalink است

آدرس Ip در لایه ی شبکه قرار دارد.

در لایه ی Transport شماره پورت قرار دارد.

یعنی آدرس در سه لایه قرار دارد

آدرس Mac مثل آدرس محله

آدرس آی پی مثل آدرس ساختمان است

شماره پورت مثل شماره واحد است . و مشخص می کند کدام پورت برای کدام پروسه است.

هر برنامه به یک موجودیت در لایه ی انتقال متصل (Bind) کرد. بنا بر این نیاز به آدرس دهی در لایه ی انتقال هست چون چندین پروسه اجرا هستند و باید مشخص شود که بسته باید به کدام پروسه تحويل داده شود اما نیاز به آدرس دهی در لایه ی Application نیست.

کار آدرس Ip مسیر یابی و هدایت بسته در شبکه توسط دستگاه های میانی (Router) است.

مک آدرس برای هدایت گام به گام یا Hop By Hop است اگر ارتباط ما یک ارتباط باشد مثل ارتباط بین هاست ها به سوییچ ها و یا هاست به روتر ها نیاز به Point To Point Point To Point آدرس Mac نداریم اما چون در شبکه توپولوژی های مختلفی داریم و نیست مثل هاب نیاز به Mac Address داریم . پس وقتی شبکه ای بیش از دو نود داریم مثال توپولوژی باس رینگ هاب و وايرلس آدرس فیزیکی نیز نیاز است .

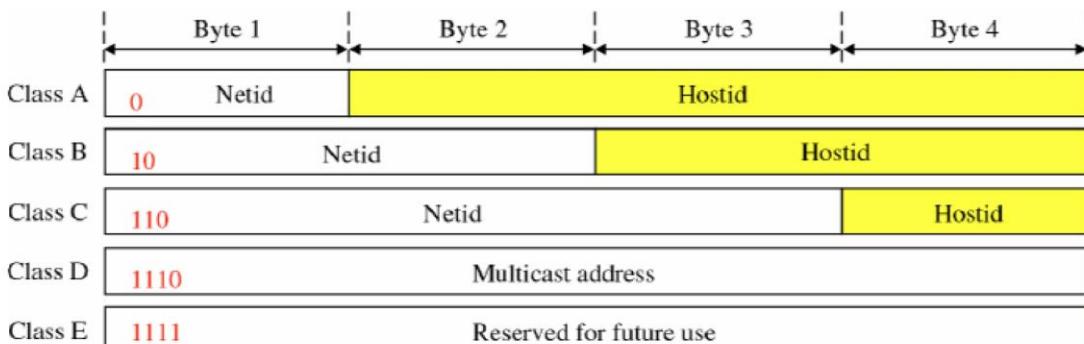
ما در لایه ۲ هم مسیریابی داریم و برای اینکه به فمد باید بسته را به کدام پورت بفرستد  
این کار با آدرس فیزیکی صورت میگیرد

## آدرس های IP

### ۰ چهار بایتی

### ۰ آدرس های classfull

### ۵ کلاس آدرس



12 of 28

آدرس های Ip دارای ۴ باید عدد هستند .

یک نوع از آدرس های Ip از نوع Classfull هستند که قدیم بیشتر استفاده میشدند و الان کمتر استفاده میشوند

۵ کلاس آدرس دارم : آدرس های کلاس Multicast برای D استفاده میشدند یعنی برای انتقال بسته به چندین هاست به صورت همزمان .

آدرس های کلاس E بجز حالتی که تمام بیت های یک است آدرس های رزرو شده اند تغیریا اصلا دیگه استفاده نمی شوند

کلاس C – B – A برای Unicast یا ارسال از یک هاست به هاست دیگر استفاده می‌شوند.

آدرس های Ip دو بخش دارند : آدرس شبکه و آدرس هاست (Net Id – Host Id) و دلیل این تقسیم بندی مسیر یابی ارزانتر و آسانتر است.

بخش Net Id در کلاس A هفت بیت است

Net Id : آدرس شبکه را مشخص می‌کند. قسمت Host Id آدرس هاست در شبکه را مشخص می‌کند. هر پورت روتر یک آدرس Ip دارد روتر ها وقتی ببینند Net Id خودش با بسته دریافتی یکی باشد متوجه می‌شود بسته برای شبکه ایست که که به پورت خودش متصل است در غیر این صورت متوجه می‌شود که بسته را باید به مقصد هدایت کند که از هفت بیت اول در کلاس A استفاده می‌کند که باعث می‌شود جدول مسیر یابی حجم کمتری داشته باشد و سپس با Host Id بسته به دست مقصد میرسد .

در کلاس A Ip ۲۸۲۴ ۲۸۲۴ هاست داریم یعنی ۱۶ میلیون هاست میتوانیم داشته باشیم .

و همچنین ۱۲۸ شبکه میتوانیم داشته باشیم .

در کلاس B سایز شبکه کمی کوچکتر و کلاس C برای شبکه های خیلی کوچک است

## آدرسهای classfull

- کلاس A : ۱۲۷ شبکه، ۱۶ میلیون میزبان
- کلاس B : ۱۶۳۸۲ شبکه، ۶۵۵۳۶ میزبان
- کلاس C : ۲ میلیون شبکه، ۲۵۶ میزبان
- کلاس D : کاربرد چندپخشی
- کلاس آدرس از روی آدرس قابل تشخیص است

### • آدرس های خاص:

- آدرس شبکه **HostId=0**
- همه بیت های **HostId = ۱** آدرس همه پخشی در زیرشبکه
- **loop back 127.x.y.z** آدرس
- آدرس جایگزین برای ارسال پیام به **0.0.0.N** در درون یک زیرشبکه

آدرس های خاص:

آدرس هاست صفر باشد غیر مجاز است.

## آدرس های بدون کلاس CIDR

- مشکل تعداد میزبان ها
- ایجاد بلاک هایی با طول متفاوت
- دو بخش: آدرس زیرشبکه (host) و آدرس میزبان (subnet)

### • ماسک زیرشبکه subnet mask

- IP: **192.168.64.0**
- Subnet mask: **255.255.255.0**

### • استفاده از نماد /

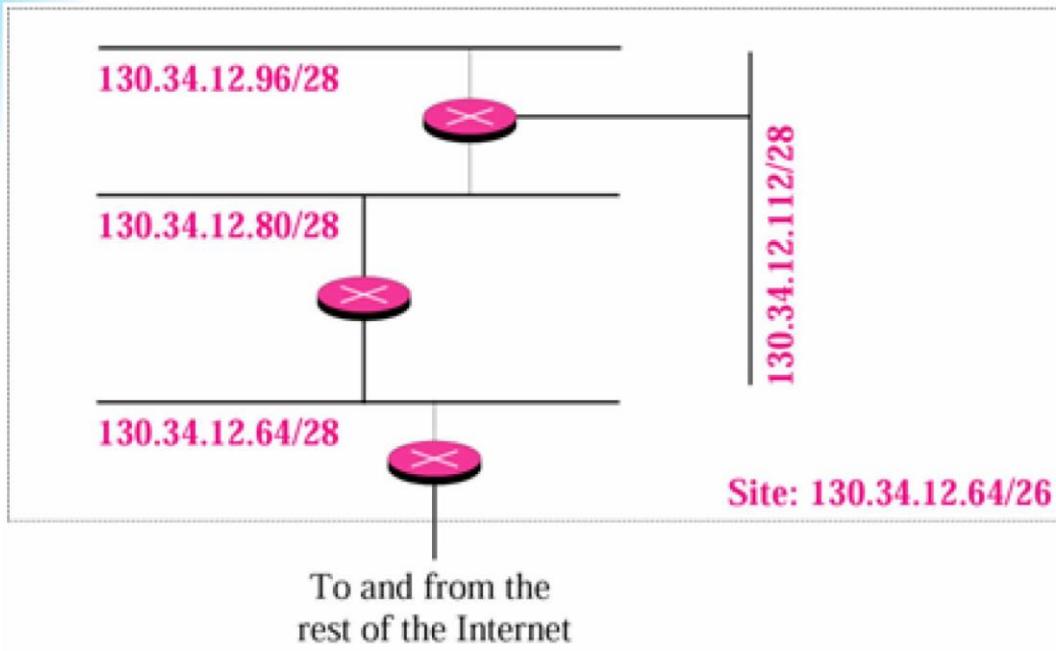
- **192.168.64.0/24**

برای جلو گیری از حذر رفتن آی پی به علت کمتر بودن تعداد هاست و یا شبکه از آدرس های بودن کلاس یا Class Less استفاده میکنیم (Cidr) در آدرس های کلاس فول در کلاس A هشت بیت اول مرز بین آدرس شبکه و آدرس میزبان است در کلاس B ۱۶ بیت اول و در کلاس C ۲۴ بیت اول مرز است اما در آدرس های کلاس لس مرز متغیر است و میتواند هر جایی باشد برای مشخص کردن این مرض از Subnetmask استفاده میکنیم در مثال بالا ۲۴ بیت اول آدرس شبکه و ۸ بیت آخر آدرس هاست هست

اگر به جای ۲۵۵ آخر ۲۵۴ بگذاریم ۲۳ بیت برای آدرس شبکه و ۹ بیت برای آدرس هاست قرار میگیرد.

همچنین متوان به شکل اسلش نیز نشانداد که در مثال بالا یعنی ۲۴ بیت اول زیر شبکه و ۸ بیت بعدی برای میزبان هست

## آدرس های بدون کلاس CIDR



در مثال بالا اسلش ۲۸ خورده یعنی ۲۸ بیت از ۳۲ بیت میشود آدرس زیر شبکه و ۴ بیت باقی می ماند و میتوان با این ۴ بیت ۱۶ هاست را آدرس دهی کرد.

در کل سایت که اسش ۲۶ خورده که ۶ بیت هاس داریم یعنی ۲<sup>۶</sup> عدد هاست داریم یعنی ۶۴ عدد هاست در کل سایت.

## آدرس های بدون کلاس CIDR

- مثال: بلاک آدرس یک ISP به شکل **190.100.0.0/16** است. این فضای بین سه گروه زیر تقسیم کنید:
- گروه ۱: ۶۴ مشتری هر کدام ۲۵۶ میزبان
- گروه ۲: ۱۲۸ مشتری هر کدام ۱۲۸ میزبان
- گروه ۳: ۱۲۸ مشتری هر کدام ۶۴ میزبان

## جواب تمرین:

گروه ۱: در این گروه هر مشتری نیازمند ۲۵۶ آدرس است. این به معنی طول  $8$  (۲۵۶=۲<sup>۸</sup>) برای پیشوند است.  
بنابراین طول پیشوند برابر است با  $۳۲-۸=۲۴$

01: 190.100.0.0/24 → 190.100.0.255/24  
02: 190.100.1.0/24 → 190.100.1.255/24

.....  
64: 190.100.63.0/24 → 190.100.63.255/24

Total =  $64 \times 256 = 16,384$

گروه دوم: در این گروه هر مشتری نیازمند ۱۲۸ آدرس است. این به معنی طول  $7$  (۱۲۸=۲<sup>۷</sup>) برای پیشوند است.  
بنابراین طول پیشوند برابر است با  $۳۲-۷=۲۵$

01: 190.100.64.0/25 → 190.100.64.127/25  
02: 190.100.64.128/25 → 190.100.64.255/25

.....  
128 190.100.127.128/25 → 190.100.127.255/25

Total =  $128 \times 256$

گروه سوم: در این گروه هر مشتری نیازمند ۶۴ آدرس است. این به معنی طول  $6$  (۶۴=۲<sup>۶</sup>) برای پیشوند است. بنابراین طول پیشوند برابر است با  $۳۲-۶=۲۶$

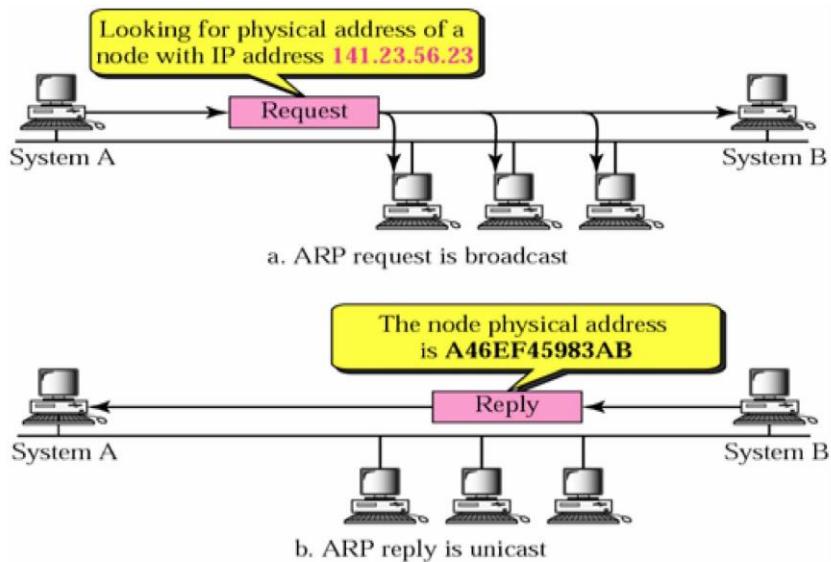
001: 190.100.128.0/26 → 190.100.128.63/26  
002: 190.100.128.64/26 → 190.100.128.127/26

.....  
128: 190.100.159.192/26 → 190.100.159.255/26

Total =  $128 \times 64 = 8,192$

## پیدا کردن یک آدرس

- پروتکل ARP
- ترجمه آدرس از آدرس شبکه به آدرس MAC



17 of 28

هر هاست وقتی می خواهد بسته ای را در شبکه ارسال کند هر لایه هدرو مخصوص خود را به بسته اضافه می کند .

برقراری ارتباط یک برنامه با یک برنامه فقط از طریق آدرس Ip و Port Number است .

برای اینکه فقط با دانستن آدرس Mac مقصد آدرس Ip را نیز بدست بیاوریم به پروتکل Arp اینکه کار ترجمه آدرس Ip را به Mac میدهد . این کار را نیز با استفاده از پروتکل Request-Reply انجام می دهد . به این صورت که یک Arp -Request می کند . آنهایی که آدرس Ip خود را در بسته نمی بینند بسته را حذف یا Arp - Reply می کنند و کامپیوتری که آدرس خود را در بسته می بیند Discard می فرستد که آدرس Mac خود را در آن اعلام می کند .

در صورتی که سیستم A یک Arp Request ارسال کند سیستم هایی که این بسته برایشان میشود آدرس Ip و Mac سیستم A را در Arp Table خود ذخیره میکنند تا در صورت نیا مک آدرس برای ارسال داده اول Arp Table را چک کنند تا اگر آدرس مک در آن موجود بود دیگر Broad Cast نکنند و ترافیک اضافی ایجاد نکنند.

موقعیت است چون Ip ها ممکن است تغییر کنند برای همین Arp Table زمان بندی شده است. و باید Refresh شود. و اگر Ip جدید توسط Arp دیده شود سطر آپدیت میشود.

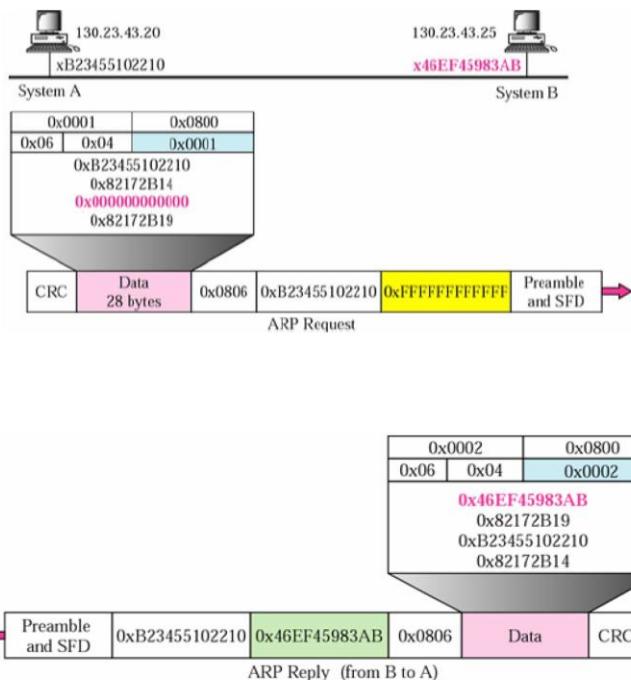
فیلد های مختلف مشخص است.

Reply در Request Target Hardware Address مشخص می شود.

## پیدا کردن یک آدرس

### • پروتکل ARP

### • مثال :



19 of 28

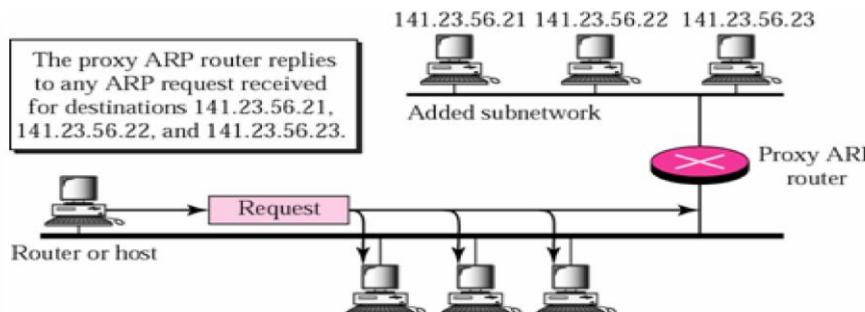
یک نمونه مثال که سیستم A قصد ارسال بسته به سیستم B را دارد اما آدرس مک را ندارد.

سیستم A یک ARP – Request به آدرس **ffffffffff** برداشت میکند و همهی هاستها آن را دریافت میکنند. بعد سیستم B که آدرس خود را در آن میبیند به آن پاسخ میدهد و آدرسش را میفرستد. مک مقصد که برآکست است (فیلد زرد رنگ) فیلد کناری مک مبدا است.

قسمت پایینی هم Reply است که در قسمت سبز رنگ میتوانیم مک سیستم B را میبینیم

## پیدا کردن یک آدرس

### • پروتکل ARP Proxy ARP router •

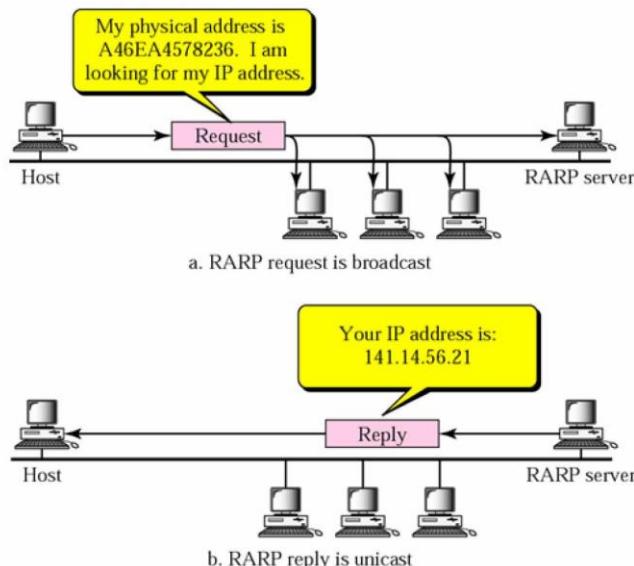


حال اگر بسته ما بخواهد به آدرسی خارج از شبکه ارسال شود . اگر Arp Request ارسال شود از روتر عبور نمی کند . حتی اگر Mac هم طوری متوجه شود باز بسته از شبکه عبور نمی کند و گام بعدی روتر است و بسته توسط روتر باید مسیر یابی شود. روتر مثل پروکسی عمل میکند اگر این روتر Ip مقصد در شبکه اش باشد (مستقیم به روتر متصل باشد) روتر Mac آدرس خودش را به جای میزبان اعلام می کند . و در این صورت عملیات هدایت یا Arp Request توسط روتر انجام میشود تا بسته به مقصد برسد.

## پیدا کردن یک آدرس

### • پروتکل RARP

### • ترجمه آدرس از آدرس MAC به آدرس شبکه



21 of 28

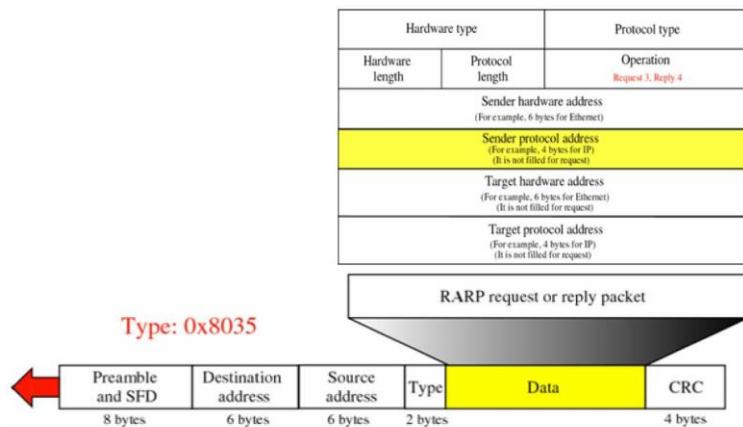
Reverse Arp یا آرپ معکوس حالت بر عکس آرپ یعنی تبدیل و ترجمه Mac به Ip است.

چون ممکن است Ip تغییر کند ممکن است حالتی رخ دهد که دستگاه Ip خود را بعد از خاموش روشن شدن فراموش کند (حافظه دائمی ندارد مثلاً) و دیگر Ip را نمی‌داند. درخواست Arp را میدهد تا Ip خود را با اعلام مکش از دیگران بپرسد.

در Dhcpc می‌توان به یه مک یک آیپی مشخص دهد و متواند رندوم هم باشد.

## پیدا کردن یک آدرس

- پروتکل RARP
- فرمت بسته ها



تصویر هم فرمت بسته Reverse Arp را نمایش میدهد که خیلی مهم نیست.

## جلسه ی چهارم :

### مسیریابی

- پیدا کردن مسیر بسته های به سمت مقصد
- انواع

- مسیریابی محلی (ستنی)
- مسیریابی سراسری (SDN)

### • متدها

- مسیریابی گام بعدی
- مسیریابی بر اساس شبکه
- مسیریابی بر اساس میزبان
- مسیریابی پیش فرض

مسیریابی پیدا کردن مسیر برای بسته ها به سمت مقصد:

وقتی روتر یک بسته را دریافت می کند باید آن را به پورت های دیگر خود هدایت کند این که چگونه باید هدایت کند با استفاده از الگوریتم مسیریابی میتوان فهمید.

پیدا کردن مسیری به سمت مقصد مشخص را مسیریابی می گوییم.

#### انواع مسیریابی:

مسیریابی محلی یا لوکال؛ در این روش روترا خودشان الگوی مسیریابی را اجرا می کنند و برای هر مقصد خاصی یک مسیر را مشخص می کنند و برای هر مقصد خاصی پورت خروجی را مشخص می کنند اینها را در یک جدول نگهداری می کنند وقتی که بسته به سمت آن مقصد آمد بسته را با استفاده از آن جدول مسیریابی به سمت مقصد ورودی پورتی که به سمت مقصد است هدایت می کنند.

( Software Define Network ) یا Sdn یا Global مسیریابی سراسری یا

روترها و وضعیت خود را به یک هاست یا Device مرکزی اعلام می کنند و آن Device مرکزی با دانستن تopoلوجی شبکه و پارامترهای دیگر مثل ترافیک روی لینک ها یا قطع شدن لینک ها و غیره برای هر کدام از روتراها تصمیم می گیرد که مسیر به سمت مقصد مشخص چی باشد، سپس این تصمیم را به هر کدام از روتراها اعلام می کند و جدول مسیریابی را به این شکل پر می کند و بعد وقتی بسته ای به سمت مقصد خاصی به دست روتر رسید روتر باید به وسیله جدول مسیریابی که سیستم مرکزی تصمیم گرفته بسته را هدایت کند.

انتخاب روش مسیریابی بستگی به شرایط و کاربرد شبکه دارد و هم می توان از مسیریابی سراسری و یا مسیریابی محلی استفاده کرد.

#### متدهای مسیریابی:

۱- متد گام بعدی.

۲- مسیریابی بر اساس شبکه در مسیریابی بین شبکه ها بیشتر استفاده می شود.

۳- مسیریابی بر اساس میزبان.

۴- مسیریابی پیش فرض.

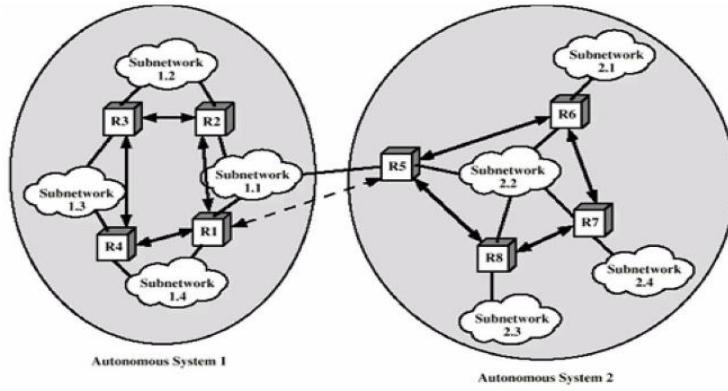
## پروتکل های مسیریابی

۰ هر بخش از اینترنت : سیستم مستقل AS

۰ انواع مسیریابی :

۰ مسیریابی داخل AS : پروتکل های RIP و OSPF

۰ مسیریابی بین AS : پروتکل BGP



26 of 28

اینترنت: شبکه ای از شبکه ها است، یعنی یک تعدادی شبکه هستند که اینها به همدیگر وصل شدند هر کدام از این شبکه ها که بخشی از اینترنت را تشکیل می دهند به آن یک سیستم مستقل

(AS) یا خود مختار می گوییم، یعنی تصمیمات داخلی اش را خودش میگیرد و وابسته به بقیه بخش های شبکه نیست اما با بقیه شبکه ها ارتباط دارد و اگر این ارتباط نباشد ما اینترنت نداریم.

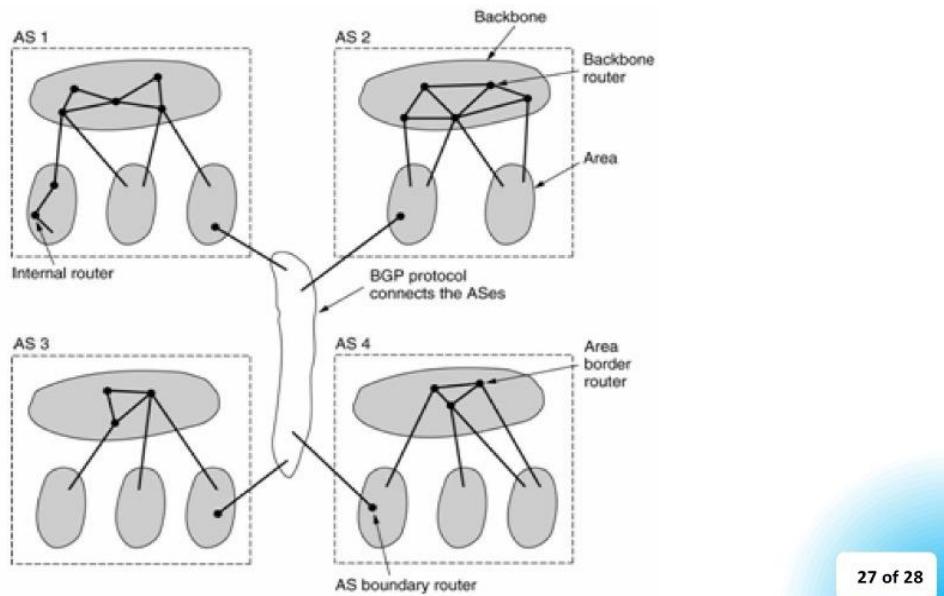
با توجه به اینکه شبکه های مختلف به هم وصل هستند انواع مسیریابی های مختلف را می توانیم داشته باشیم.

یک سری از مسیرها مخصوص داخل AS هستند و برای مسیریابی درون AS استفاده می شوند، پروتکل های مثل Ospf و Rip برای درون AS و بین هاست ها و روترا های داخل یک شبکه استفاده می شود.

برای مسیریابی بین AS ها پروتکلی مثل Bgp بیشتر مورد استفاده قرار می گیرد. حالا روش مسیریابی می تواند سنتی باشد که این پروتکل ها هستند یا می توانند روش مسیریابی است Sdn باشد که بین AS کار راحتی نیست و عموماً انجام نمی شود، اما درون AS از Sdn می توان استفاده کرد.

## پروتکل های مسیریابی

### ۰ رابطه بین AS ها و ستون فقرات



در شکل ۴ تا AS نشان داده شده است، در هر کدام از AS ها یکسری روتر است که به هم وصل هستند و یک سری روتر هست که به این روترا ها هم میتوانند هاست ها وصل شوند ولی ارتباط

اصلی و پایه ای داخل AS بر عهده یک سری از این روترها است، یعنی هر کدام از این روترها خراب شود ضربه بدی به ارتباطات درون AS وارد میشود؛ درسته ممکن است با حذف یکی از این روترها یا لینک‌ها ارتباط بین هاست‌های داخل شبکه قطع نشود ولی بخش عظیمی از ترافیک آن روتر یا لینک می‌تواند هندل کند بر عهده روترهای دیگر می‌افتد و ارتباط ضعیف خواهد شد؛ به این روترهایی که بیشتر ترافیک از آنها عبور می‌کند و ارتباط‌های پایه و اصلی در AS هستند بهش Backbone یا ستون فقرات می‌گوییم و این روترهایی که برای دسترسی استفاده می‌شوند اهمیت کمتری دارند و اگر خراب شود کل شبکه سرپا هست و اتفاق بدی برای شبکه نمی‌افتد.

این AS‌ها با استفاده از روترهای مرزی(Boundary Router) می‌توانند به هم‌دیگر متصل شوند.

## پروتکل‌های مسیریابی

مسیریابی ثابت  
مسیریابی سیل آسا  
مسیریابی اتفاقی  
مسیریابی تطبیقی  
مسیریابی مبدا

انواع پروتکل‌های مسیریابی سنتی که استفاده می‌شود.

شامل:

- مسیریابی ثابت.
- مسیریابی سیل آسا.
- مسیریابی اتفاقی.
- مسیریابی تطبیقی.

- مسیریابی مبدا.

## پروتکل های مسیریابی

• **Autonomous system :** شبکه ای که توسط یک سازمان مدیریت و نگهداری می شود.

• اینترنت : مجموعه ای از AS های متصل

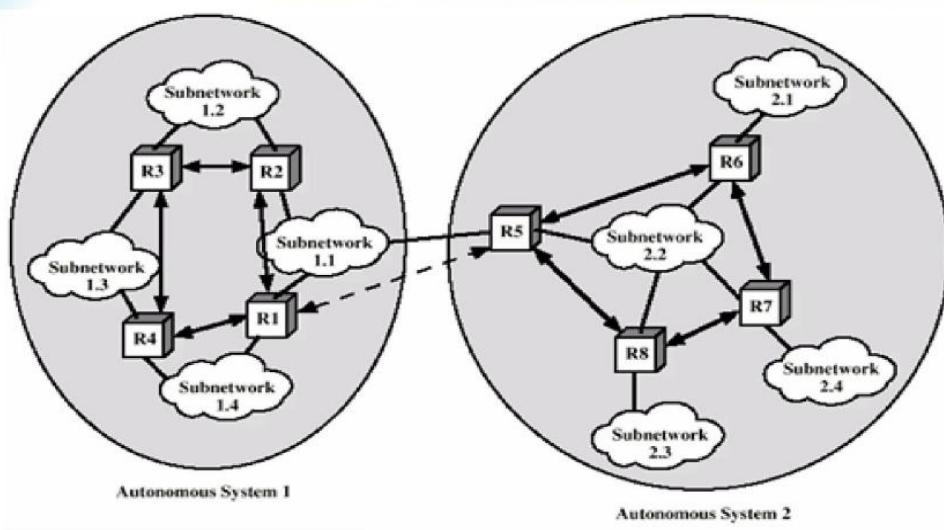
• انواع مسیریابی در اینترنت:

• مسیریابی داخلی (Interior Gateway Protocol IGP) : مسیریابی درون AS ها، RIP و OSPF

• مسیریابی بیرونی (Exterior Gateway Protocol EGP) : مسیریابی بین AS ها، BGP

• مثل شبکه سازمان تامین اجتماعی یا شبکه شهرداری و ..

## پروتکل های مسیریابی



Interior router protocol →→→  
Exterior router protocol → - - - →

همانطور که در شکل می بینید در ناحیه AS پروتکل درونی داریم و بین ASها همانطور که با فلش خط چین نشان داده شده مسیریابی بیرونی داریم.

## پروتکل های مسیریابی

- **Store & foreword :** ذخیره کامل بسته، تعیین گام بعدی و ارسال هنگام آزاد بودن خط خروجی
- **Next hop forwarding :** تنها ارسال به مسیریاب بعدی، سایر مسائل مثل زمان رسیدن بسته به مقصد اهمیت ندارد
- استقلال از مبدأ : در مسیریابی مبدأ مهم نیست (در مدار مجازی مبدأ هم مهم است)

چند تا از خصوصیت های پروتکل های مسیریابی را داریم

- پروتکل های مسیریابی که در شبکه های کامپیوتری استفاده می شوند Store & Forward هستند؛ یعنی بسته اول در روتر به طور کامل ذخیره می شود و بعد راجع به آن تصمیم می گیرد و آن را در حافظه داخلی خود نگه می دارد تا آن را به پورت خروجی ارسال کند. اگر آن پورت مشغول ارسال پکت دیگری باشد باید صبر کنند و در صف قرار بگیرد تا نوبت شود و روی پورت خروجی ارسال شود.
- مشخصه بعدی Next Hop Forwarding است؛ که دغدغه روتر ما این است که بسته را برای روتر بعدی بفرستد و مثلاً زمان رسیدن بسته به مقصد اهمیت ندارد.
- استقلال از مبدأ در مسیریابی مبدأ مهم نیست، یعنی بسته از کجا آمده اهمیتی نداره؛ اما این که بسته به کجا می خواهد برود مهم است.

در مدار مجازی مبدا مهم است؛ یک کانکشن بین مبدا و مقصد قبل از ارسال پکت برقرار می شود و در این کانکشن مسیر مشخص می شود ممکن است یک بسته به یک مقصد مشترک اما از دو مبدا مختلف برسد.

## پروتکل های مسیریابی

### ۰ مسیریابی در راهگزینی بسته ای

۰ نیازمندی ها : کار کرد صحیح، سادگی، مقاوم بودن، پایداری، عدالت، بهینگی، کارایی

۰ معیارهای کارایی : تعداد گام، هزینه، تاخیر، گذردگی

۰ زمان تصمیم گیری : رسیدن بسته (در **datagram**) یا ایجاد ارتباط (در **virtual circuit**)

۰ محل تصمیم گیری : هر گره، گره مرکزی، گره مبدا (**source routing**)

۰ جمع آوری اطلاعات : نیازی نیست (مسیریابی ایستا)، محلی، همه گره ها، گره های همسایه

۰ زمان به روز رسانی : پریودیک، با ایجاد تغییر مانند تغییر بار یا توپولوژی

می خواهیم در پکت سوئیچینگ ببینیم که مسیریابی چگونه است:

اول باید ببینیم که در مسیریابی چه نیازمندیهایی داریم و چه پارامترهایی را باید در نظر بگیریم.

- کار کرد صحیح یعنی اینکه روترا بسته را به درستی هدایت کند.
- سادگی یعنی هر چه پروتکل ساده تر باشد، بهتر است و پاسخ کل پیچیده نیاز به سخت افزار پیچیده تری دارد و در خیلی از روتراها ممکن است وجود نداشته باشد.
- مقاوم بودن یعنی با تغییرات یک شبکه همچنان شبکه به درستی کار کند .
- پایداری یعنی ما به یک مسیر نهایی دست پیدا کنیم مسیر را با گذشت زمان تغییر ندهیم.
- عدالت یعنی بسته های به مقاصد مختلف را به موقع Forward کند.

- بهینگی یعنی مسیری که برای یک مقصد مشخص انتخاب می شود بهترین مسیر باشد.

#### معیارهای کارایی:

- تعداد گام یعنی تعداد روترهایی که در یک مسیر بسته از آنها عبور می کند حداقل باشد.
- هزینه یعنی ممکن است در یک شبکه مسافتی که بسته طی می کند باشد یا نرخ بیت مسیرهایی که از آنها عبور می کند باشد.
- تاخیر یعنی ممکن است ما بسته را از یک مسیری هدایت کنیم که تعداد گامهای کمی دارد اما تاخیر آن زیاد است و بر عکس مسیر دیگری ممکن است وجود داشته باشد که تعداد گامهای بیشتری داشته باشد اما تاخیر کمتری داشته باشد به خاطر کمتر شلوغ بودن شبکه وجود لینکهای پر سرعت در آن مسیر.

#### زمان تصمیم گیری:

در پکیج سوئیچینگ نتورک‌ها تصمیم گیری برای یک بسته زمانی انجام می‌گیرد که آن بسته به روتر برسد، پس هنگام رسیدن بسته به روتر تصمیم گیری انجام می‌شود، روتر جدول مسیر یابی خود را بررسی می‌کند و بررسی می‌کند که بسته را به کدام پورت‌ش بفرستد. دیتاگرام همان راه گزینی بسته‌ای است.

#### زمان تصمیم گیری:

می‌تواند زمان ایجاد ارتباط باشد که در Virtual Circuit داریم اول کانکشن ایجاد می‌کنیم، بعد بسته را ارسال می‌کنیم و در زمان ایجاد کانکشن هست که تصمیم می‌گیریم که بسته از کدام مسیر و از کدام روترهای عبور کند.

#### محل تصمیم گیری:

در روش‌های مسیریابی سنتی محل تصمیم گیری برای هر نود همان نود است و هر روتر خودش تصمیم می‌گیرد که بسته را به کدام پورت ارسال کند.

در Sdn گره مرکزی این تصمیم را می گیرد.

گره مبدا یعنی مبدا تصمیم می گیرد که بسته را از کجا هدایت کند تا به مقصد برسد.

### جمع آوری اطلاعات:

در بعضی از الگوریتم های مسیریابی نیازی به جمع آوری اطلاعات نیست مثلاً در مسیریابی ایستانا نیاز این است که اطلاعاتی از شبکه جمع آوری بشود چون مسیر هیچ وقت تغییر نمی کند مگر اینکه مدیر سیستم مسیرها را تغییر دهد.

محلى یعنی هر نود فقط از نودهای مجاورش اطلاعات دارد و بر اساس همانها تصمیم گیری می کنند.

همه گره ها مثل Sdn که اطلاعات تمام روترا را دریافت میکنند. همسایگی یعنی هر نود از همسایه های خودش اطلاعاتی را دریافت کند و بر اساس آنها تصمیم گیری کند.

زمان به روز رسانی یعنی کی ما مسیرها را تغییر دهیم.

- می تواند پریودیک باشد یعنی مثلاً هر ۵ دقیقه یک بار مسیرها آپدیت شود.
- زمان به روز رسانی با ایجاد تغییر باشد مثلاً تغییر بار شبکه یا توپولوژی.

## پروتکل های مسیریابی

• مسیریابی : ایستا یا پویا

• مسیریابی ایستا

• مسیریابی ثابت

**Source routing .**

• مسیریابی سیل آسا

• مسیریابی اتفاقی

• مسیریابی تطبیقی

• مسیریابی مبتنی بر جریان

مسیریابی میتواند ایستا یا پویا باشد

مسیریابی ایستا مسیریابی ثابت یا Source Routing است

## پروتکل های مسیریابی

• مسیریابی ثابت

• مسیر ثابتی برای هر زوج گره

• تغییر مسیر نداریم. حداکثر با تغییر توپولوژی شبکه

• در هر گره برای هم مقصد : فقط گره بعدی

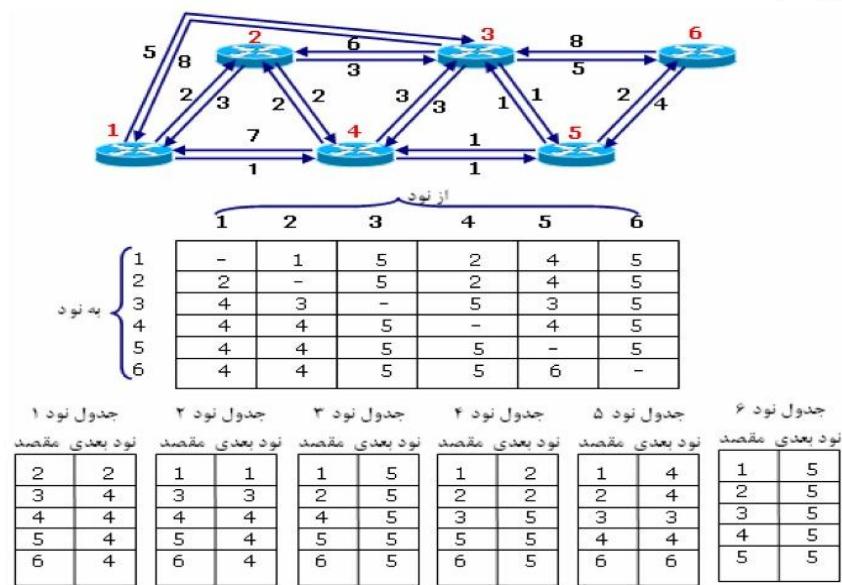
• دایرکتوری گره : گره بعدی برای هر مقصد

در مسیر یابی ثابت، برای هر دوتا گره که در شبکه وجود دارد یعنی از مبدا تا مقصد یک مسیر ثابت در نظر میگیریم و تغییر مسیر نداریم مگر اینکه تغییر توپولوژی شبکه را داشته باشیم، یعنی یک روتر حذف یا اضافه شود.

در هر گره برای هر مقصد ما در جدول مسیریابی فقط گره بعدی را لازم است مشخص کنیم بنابراین یک جدولی که بهش دایرکتوری گره می‌گوییم، ایجاد می‌شود که گره بعدی برای هر مقصد باید مشخص باشد؛ در جدول مسیریابی مشخص می‌شود که روتر هر بسته‌ای که دریافت کرد را باید باهاش چه کار کند.

## پروتکل‌های مسیریابی

### • مسیریابی ثابت



یک مثال از مسیریابی ثابت است، شبکه با شش گره است که هزینه هر لینک مشخص است

در جدول Next Hop را برای هر نод مشخص کرده مثلاً از ۶ به ۴ Next Hop است این جدول را مدیر شبکه دارد و در پایین برای هر کدام از گره‌ها این جدول مسیریابی را تهیه کردند

## پروتکل های مسیریابی

### • مسیریابی سیل آسا

• ارسال بسته ها به همه گره های دیگر

• ایجاد بار زیاد در شبکه

• ایجاد بسته های تکراری

• توقف ارسال:

• استفاده از شناسه بسته و عدم ارسال بسته تکراری

• یا استفاده از TTL

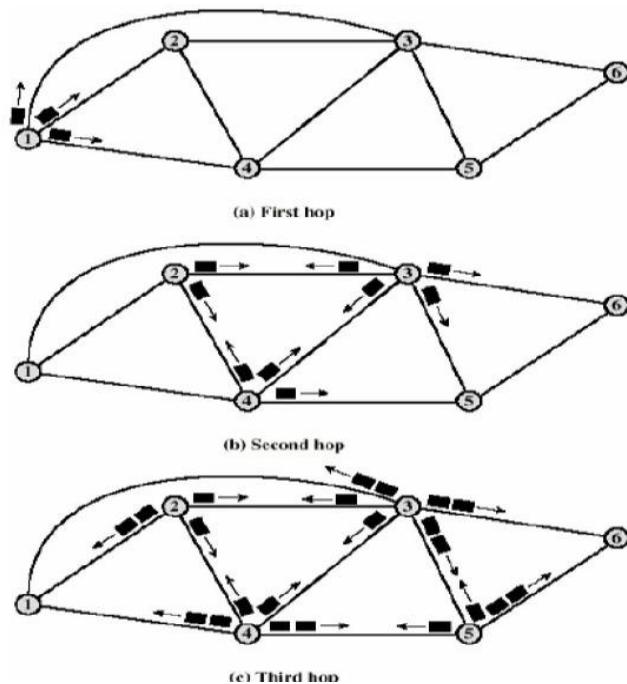
در مسیریابی سیل آسا بسته به همه گره های دیگر ارسال میشود؛ یعنی اگر یک گره ای پکت دریافت کرد آن را به همه گره ها ارسال می کند، این کار باعث ایجاد بار زیادی در شبکه میشود زیرا ما هر بسته را چند بار توسط هر روتر ارسال میکنیم و بسته های تکراری زیادی به وجود می آید Forwarding باید یک جایی متوقف شود که توقف ارسال های مجدد به دو شکل انجام می شود:

وقتی یک بسته را دریافت می کنیم شناسه آن را یادداشت کنیم دوباره اگر از مسیر دیگری آن بسته را دریافت کردیم دیگر آن را ارسال نکنیم.

با استفاده از Ttl یعنی بسته زمانی که به نظر می رسد شمارنده Ttl آن یک واحد کم شود و وقتی Ttl آن به صفر رسید دیگر ارسال نشود مقدار Ttl را باید به اندازه دورترین فاصله بین روتراها در شبکه بگذاریم تا مطمئن شویم پکت به مقصد می رسد.

## پروتکل های مسیریابی

### • مسیریابی سیل آسا



13 of 22

در مرحله اول گره یک پکت را به همه لینک هایش می فرستد یعنی گره های دو، سه و چهار پکت را دریافت می کنند، در مرحله دوم گره دو به همه پورت ها به جز پورتی که پکت را دریافت کرده، بسته را ارسال می کند ۳ و ۴ هم به همین صورت...

## پروتکل های مسیریابی

### ۰ خواص مسیریابی سیل آسا

۰ آزمودن همه مسیرها : عدم اهمیت خرابی یک اتصال یا خروجی گره

← یک مسیریابی قوی : قابل استفاده در کاربردهای حساس و ارسال پیامهای اضطراری، مانند کاربردهای نظامی

۰ رسیدن حداقل یک کپی به مقصد، اولین کپی = کوتاه ترین مسیر

← قابل استفاده در بروایی مدار مجازی، یا در مسیریابی پویا

۰ ملاقات همه گره های شبکه

← انتشار داده به همه گره ها، یا انتشار اطلاعات مسیریابی

مسیریابی سیل آسا علاوه بر مشکلاتی که دارد یکسری خواص و مزایا نیز دارد:

همه مسیرها طی می شوند اگر در شبکه یک روتور خراب شده باشد، اگر مسیریابی سیل آسا نداشته باشیم تا این خرابی به وسیله بقیه روتورها تشخیص داده شود و مسیر جایگزین برای ارتباط به مقصد مشخص ایجاد شود، یک تعدادی از بسته ها سرگردان می مانند و حذف می شوند ولی در مسیریابی سیل آسا این اتفاق نمی افتد و خرابی یک اتصال تاثیر زیادی در رسیدن بسته به مقصد ندارد به شرطی که آن خرابی باعث دو قسمت شدن شبکه نشود، پس با توجه به اینکه این روش خیلی ساده هست اما بسیار قوی هست و استفاده آن در کاربردهای حساسی مثل نظامی یا ارسال پیام های اضطراری میتواند باشد.

در این روش مسیریابی ما مطمئن هستیم که حداقل یک کپی از بسته به مقصد می رسد و اولین کپی از طریق کوتاه ترین مسیر به مقصد می رسد، برای ایجاد مدار مجازی می توان از این روش استفاده کرد تا کوتاهترین مسیر را بتوانیم پیدا کنیم.

بسته ارسالی به همه گره های شبکه خواهد رسید و همه گرهها را ملاقات می کند و این برای Broadcast کردن دیتا می تواند استفاده شود.

## پروتکل های مسیریابی

### • مسیریابی اتفاقی

• شبیه مسیریابی سیل آسا: انتخاب فقط یک خروجی به صورت تصادفی

• در صورت وزن یکسان اتصالات خروجی: می توان چرخشی عمل کرد

• احتمال می تواند بر اساس نرخ لینک ها باشد

$$P_i = \frac{R_i}{\sum_j R_j}$$

P<sub>i</sub>: احتمال انتخاب اتصال آ  
R<sub>i</sub>: نرخ انتقال داده در اتصال آ

روش بعدی مسیریابی Random است: خیلی شبیه مسیریابی سیل آسا است. اختلاف آن در این است که به جای اینکه همه خروجی ها انتخاب شوند یک خروجی به صورت تصادفی انتخاب می شود و اگر پورت های خروجی ما وزن یکسان داشته باشند می توانیم این را اتفاقی انتخاب نکنیم و به صورت چرخشی انتخاب کنیم.

احتمال انتخاب هر کدام از لینک ها بر اساس نرخ لینک می تواند باشد؛ مثلاً در رابطه احتمال انتخاب لینک I برابر است Rate I به مجموع Rate I همه لینکها در این صورت اگر یک لینکی Rate بیشتری داشته باشد احتمال انتخاب آن بیشتر خواهد شد و بسته های بیشتری به آن سمت ارسال خواهد شد.

این مسیر یابی یکسری اشکالات دارد و یک سری از مزايا هم دارد:

یکی از مزايا این است که خیلی ساده هست، نیاز به اجرای الگوریتم خاصی ندارد.

یکی دیگر از مزیت های آن این است که، مثل سیل آسا بسته های تکرار ایجاد نمی کند.

اشکال آن این است که، ممکن است بسته ای به خاطر اینکه از لینکهای رندوم استفاده می کند خیلی دیرتر به مقصد برسد یا شاید اصلاً نرسد.

## پروتکل های مسیریابی

### • مسیریابی تطبیقی

#### • تغییر تصمیمات مسیریابی بر اساس شرایط

◦ شرایط:

◦ خرابی یک گره یا لینک

◦ ازدحام بار در یک بخش شبکه

◦ نیاز به تبادل اطلاعات بین گره ها

◦ پیچیده تر شدن تصمیم مسیریابی، افزایش پردازش گره ها

◦ بار ترافیکی حاصل از اطلاعات جابجا شونده

◦ عکس العمل سریع یا کند یک استراتژی در هنگام تغییر شرایط

روش مسیریابی تطبیقی؛ که بر اساس شرایط می توانیم تصمیمات مسیریابی را تغییر بدھیم.

شرایطی که مد نظر است:

می تواند خرابی یک لینک یا گره باشد، یعنی تغییر توپولوژی

یا ازدحام بار در یک بخش شبکه باشد، یعنی یک لینک اگر تعداد بسته های دریافتی به سمت مش زیاد باشد روتر می تواند مسیر را از روی آن لینک به لینک دیگر تغییر دهد، برای این کار نیاز به

تبدال اطلاعات بین گره ها هست و گره ها باید با هم اطلاعات رد و بدل کنند که از تغییرات آگاه باشند.

این تبدال اطلاعات بین گره ها باعث پیچیده تر شدن تصمیم مسیریابی می شود. روتر ها بر اساس شبکه باید تصمیم گیری کنند، بنابراین پردازش گره ها بیشتر خواهد شد و گره ها باید قدرت پردازشی بیشتری داشته باشند.

اطلاعاتی که بین گره ها برای دانستن اطلاعات همدیگر رد و بدل می شود باعث می شود که بار ترافیکی شبکه زیاد شود، وقتی که تغییری در شبکه اتفاق می افتد بسته به اینکه چه استراتژی برای عکس العمل روی آن تغییر دهیم می توانیم عکس العمل سریع یا کند داشته باشیم.

## پروتکل های مسیریابی

- دلایل استفاده از مسیریابی تطبیقی
  - امکان بهبود کارایی
  - کمک به کنترل ازدحام
- طبقه بندی استراتژی ها مبتنی بر اطلاعات مبدأ: محلی ( فقط خود گره )، همسایه ها، همه گره ها
- مثالی از استراتژی مبتنی بر اطلاعات محلی:
  - هدایت بسته به خروجی با کمترین طول صفحه
  - نتیجه = توازن بار

چرا از مسیریابی تطبیقی استفاده میکنیم ؟

کارایی را بهتر می‌تواند کند، یعنی اگر مثلاً ما روش مسیریابی سیل آسا را داریم، می‌دانیم که ازدحام در شبکه زیاد است؛ یا مثلاً مسیریابی ایستا را داریم، تغییراتی اگر در شبکه اتفاق بیفتد روتراها به آن عکس العمل ندارند، مدیر شبکه باید این تغییرات را انجام دهد. اما در مسیریابی تطبیقی به محض ایجاد تحت تغییرات چه از نظر توپولوژی و چه از نظر بار ترافیکی مسیرهای بهتری جایگزین خواهد شد و این باعث بهبود کارایی شبکه می‌شود و همچنین به کنترل ازدحام می‌توانیم کمک کنیم، یعنی اگر بخشی از شبکه دچار ازدحام شود، مسیر می‌تواند تغییر کند و مسیر به سمتی برود که ازدحام کمتری در شبکه وجود دارد.

استراتژی‌های تغییر مسیریابی را می‌توانیم مبنی بر اطلاعات دریافتی دسته بندی کنیم، که می‌تواند اطلاعات محلی باشد؛ یعنی فقط خود گره از وضعیت صفحهای خودش برای پورت‌های مختلف برای تغییر مسیریابی استفاده کنند، یا می‌تواند اطلاعات را از همسایه هایش دریافت کند و به صورت اطراف خود گره باشد یا می‌تواند اطلاعات را از همه گرههای شبکه دریافت کند.

مثالی از استراتژی مبنی بر اطلاعات محلی:

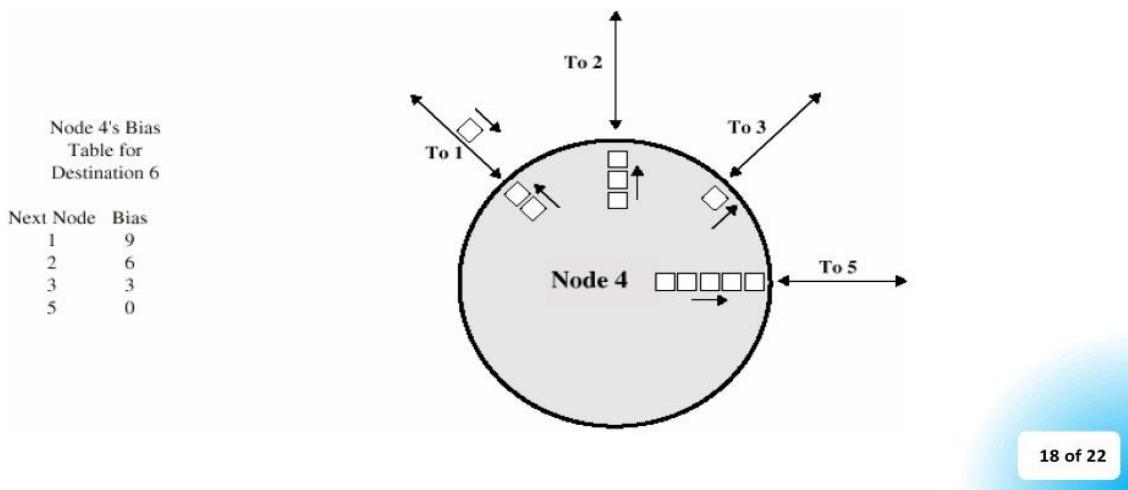
روتر بر اساس طول صفحه موجود در گرههای خروجی تصمیم بگیرد که بسته را به پورت خروجی با کمترین طول صفحه هدایت کند و نتیجه آن این است که بار ترافیکی روی لینک‌های مختلف در شبکه متوازن خواهد شد.

## پروتکل های مسیریابی

• مسیریابی تطبیقی ترکیب با مسیریابی اتفاقی

• استفاده از  $B_i$  برای هر لینک  $i$  به عنوان Bias

• ارسال روی لینکی با حداقل  $Q + B_i$



می‌توان مسیریابی تطبیقی را با مسیریابی اتفاقی ترکیب کنیم؛ در مسیریابی اتفاقی پورت خروجی را به صورت اتفاقی انتخاب می‌کنیم و اگر احتمال یکسان باشد می‌توانیم به صورت چرخشی و به ترتیب عمل کنیم و پورت خروجی را انتخاب کنیم، اگر بخواهیم مسیریابی تطبیقی را به مسیریابی Rndom اضافه کنیم یک پارامتر به اسم  $B_i$  برای هر لینک  $i$  به عنوان Bias اضافه می‌شود که  $B_i$  نشان دهنده طول صفحه را کدام از پورت‌های خروجی هست،

ارسال روی لینک با حداقل طول صفحه به اضافه  $B_i$  که هم مسیریابی اتفاقی و هم طول بسته را می‌توانیم برای ارسال بسته یک لینک خروجی اعمال کنیم.

## پروتکل های مسیریابی

### • مسیریابی تطبیقی

- استراتژی های مبتنی بر گره های همسایه یا همه گره ها متقابل تر هستند
- توزیع شده یا مرکزی باشند
- توزیع شده : ارسال اطلاعات به سایر گره ها
- مرکزی : ارسال اطلاعات به گره مرکزی

در مسیریابی تطبیقی، استراتژی های مبتنی بر دریافت اطلاعات از بقیه گره های همسایه یا همه گره های شبکه متقابل تر از اطلاعات مبتنی بر اطلاعات خود نود به تنها یی است.

مسیریابی تطبیقی می تواند توزیع شده یا مرکزی باشد؛ توزیع شده یعنی در هر مسیر یاب تصمیم گیری شود و مرکزی یعنی در نود مرکزی تصمیم گیری شود.

در روش توزیع شده، ارسال اطلاعات به سایر گره ها انجام می شود و در مرکز ارسال اطلاعات به گره مرکزی است، یعنی در روش توزیع شده هر نود اطلاعات بقیه نود ها را باید دریافت کنند اما در روش مرکزی فقط نود مرکزی هست که اطلاعات همه نود ها را دریافت کند و نتیجه را به همه نود ها اعلام کند.

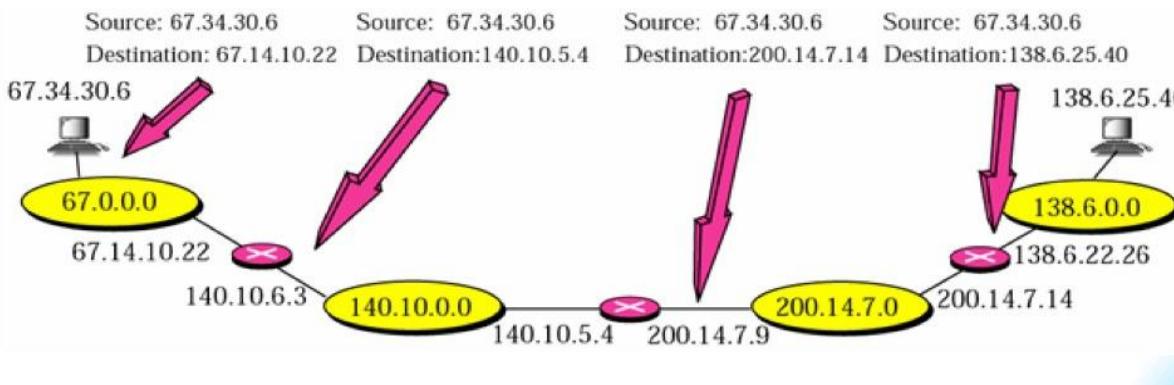
# پروتکل های مسیریابی

## • مسیریابی مبدا (source routing)

- کاربرد کم در WAN

- بیشتر کاربرد در اتصال پل ها در LAN

- در شبکه محلی token ring



مسیریابی مبدا که در مبدا تعیین می شود، که بسته از کدام روتراها باید عبور کند تا به مقصد برسد یعنی کل مسیر در مبدأ مشخص می شود و داخل بسته قرار داده می شود و روتر از خود بسته متوجه می شود که بسته را به کدام روتر باید ارسال کند.

# پروتکل های مسیریابی

## • مسیریابی مبدا (source routing)

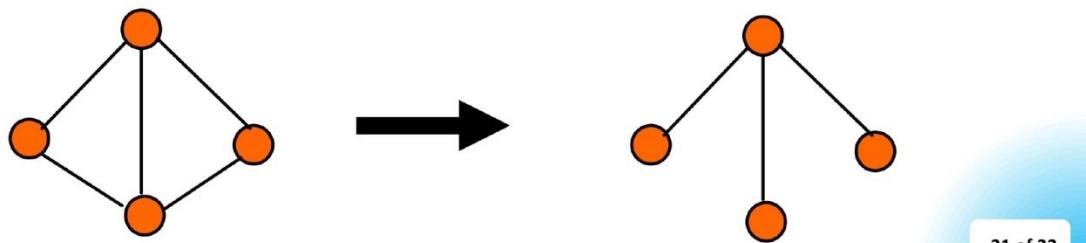
◦ کاربرد کم در WAN

◦ بیشتر کاربرد در اتصال پل ها در LAN

◦ در شبکه محلی token ring

## • مسیریابی مبدا در پل های شفاف (transparent bridges)

◦ واحد داده پروتکل پل (درخت پوشش spanning tree) : BPDU



در Wan استفاده کمتری دارد اما در اتصال پل ها در Lan بیشتر مورد استفاده قرار می گیرد، همچنین در شبکه های محلی Token Ring هم استفاده می شود.

اتصال پل ها در شبکه محلی در پلهای شفاف مورد استفاده قرار می گیرد، یک شبکه داریم که نود و رینگ های آن یک گراف را تشکیل می دهد، این گراف را با استفاده از پروتکل یا الگوریتم Spanning Tree می توانیم به یک درخت پوشش تبدیل کنیم یعنی درختی که تمام نود ها را در اختیار دارد و قاعدهاً در این درخت Loop وجود نخواهد داشت و این باعث می شود که ما بسته را از مبدأ به بقیه نود های درخت ارسال کنیم.

## جلسه پنجم:

### پروتکل های مسیریابی در شبکه

## پروتکل های مسیریابی

### • مسیریابی مبتنی بر جریان (flow based routing)

◦ جریان = میزان بار یک مسیر

◦ جریان = بسته هایی با خصوصیات مشترک

◦ دسته بندی بسته ها (packet classification) و مسیریابی برای هر دسته

◦ ملزمات:

◦ توپولوژی شبکه

◦ ماتریس ترافیک

◦ ماتریس ظرفیت

مسیریابی مبتنی بر جریان (Float Based Routing) : یک نوع مسیریابی می باشد که به جریان روی لینک ها توجه کرده و درواقع ازدحام را در نظر میگیرد.

جریان: در اینجا ما به میزان باری که روی یک مسیر قرار بگیرد **جریان** می گوئیم. به عبارت دیگر جریان را بسته هایی با خصوصیات مشترک (مبدأ و مقصد مشترک) در یک Application نیز میگویند.

دسته بندی بسته ها (**Packet Classification**) : می توان بسته های ارسالی را کلاس بندی کرد. مثل بسته هایی که نیازمند تاخیر کم هستند را در یک کلاس قرار دهیم و یا بسته های حساس به Packet Lost را در یک کلاس بیاوریم.

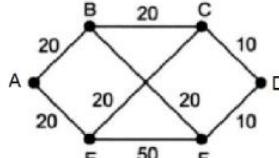
نیازمندی های مسیر یابی مبتنی بر جریان:

- ۱ - توپولوژی شبکه (می بایست حتما با ساختار توپولوژی شبکه مذکور آشنا باشیم)
- ۲ - ماتریس ترافیک (علم به میزان ترافیک عبوری از لینک ها)
- ۳ - ماتریس ظرفیت (ظرفیت یک لینک به توانایی حجم عبور دیتا از خودش می باشد)

با داشتن اطلاعات بالا می توانیم مسیر یابی مبتنی بر جریان را انجام دهیم.

برای مثال شکل زیر را در نظر بگیرید:

## پروتکل های مسیریابی



		Destination					
		A	B	C	D	E	F
Source	A	9 AB	4 ABC	1 ABCD	7 AE	4 AEF	
	B	9 BA	8 BC	3 BFD	2 BFE	4 BF	
C	4 CBA	8 CB		3 CD	3 CE	2 CEF	
D	1 DFBA	3 DFB	3 DC		3 DCE	4 DF	
E	7 EA	2 EFB	3 EC	3 ECD		5 EF	
F	4 FEA	4 FB	2 FEC	4 FD	5 FE		

i	Line	$\lambda_i$ (pkts/sec)	$C_i$ (kbps)	$\mu C_i$ (pkts/sec)	$T_i$ (msec)	Weight
1	AB	14	20	25	91	0.171
2	BC	12	20	25	77	0.146
3	CD	6	10	12.5	154	0.073
4	AE	11	20	25	71	0.134
5	EF	13	50	62.5	20	0.159
6	FD	8	10	12.5	222	0.098
7	BF	10	20	25	67	0.122
8	EC	8	20	25	58	0.098

5 of 18

ظرفیت هر لینک روی آن نوشته شده است که برای مثال AB ظرفیتش ۲۰ Kbps می باشد(سمت چپ بالا) که با آن را در جدول پایین اسلاید بالا نشان داده ایم. Demand روی آن را با لاندا بر حسب Pkts/Sec نمایش میدهیم. در جدول سمت راست Source و Destination را مشخص کرده ایم و ماتریس ظرفیت را ایجاد نموده ایم. در جدول مذکور لینک ها نسبت به عکس خود متقارن هستند. کافی است در جدول مبدا را در حروف عمودی سمت راست جدول و مقصد را در حروف افقی بالای جدول پیدا کرده و نسبت به گره ای که از آن عبور میکند به عدد مورد نظر دست یابیم. جهت محاسبه می بایست تمامی اعداد در مسیر را نیز جمع کنیم. برای مثال اگر مبدا A و مقصد F باشد در جدول مسیر AD به صورتی است که باید به ترتیب از B و سپس از C عبور کند در این حالت ما در کاتریس هر جا حروف AB را دیدیم عدد

داخل ماتریس را جمع میکنیم. پس در مسیر باید  $9 + 4 + 1$  را جمع بزنیم که معادل ۱۴ می شود و همان طور که در جدول مشاهده می کنید لاندا( $\lambda$ ) برابر ۱۴ می باشد.

## پروتکل های مسیریابی

•  $T$  : زمان سرویس

$$\bullet T = 1 / (\mu C - \lambda)$$

•  $\mu$  : نرخ سرویس

•  $\lambda$  : نرخ ورود

•  $C$  : ظرفیت لینک

• طول بسته ها را برابر  $1/\mu$  در نظر گرفته و در این مثال = ۸۰۰ بیت

•  $w$  : وزن

$$\bullet w_i = T_i / (T_1 + T_2 + \dots)$$

6 of 18

در جدول بالا یکسری پارامتر داریم که دارای یک رابطه با یکدیگر هستند:

$$T = 1 / (\mu C - \lambda)$$

•  $T$  : زمان سرویس (Service Time) : به زمان سپری شده جهت ارسال و دریافت هر پکت روی هر لینک می گویند. در واقع روی هر لینک چقدر انتظار می کشد تا وارد لینک بعدی شود را زمان سرویس می نامند.

•  $\mu$  : نرخ سرویس (Service Rate)

•  $\lambda$  : نرخ ورود

• **C:** ظرفیت لینک (Link Capacity)

• **W:** وزن (Weight)

اگر طول بسته را برابر  $\mu$  در نظر بگیریم در این مثال برابر ۸۰۰ بیت می شود. پس

نرخ سرویس ما  $1/800$  می باشد.

$$W_i = T_i / (T_1 + T_2 + \dots)$$

در نهایت هرمسیر که وزن بیشتری داشته باشد جهت سرویس انتخاب می شوند. زیرا آن مسیر بسته ها را سریع تر ارسال می کند و بار کمتری دارد.

## الگوریتم های مسیریابی در اینترنت

• **بردار فاصله Distance Vector**

◦ اینترنت اولیه، شبکه های ناول و DEC

◦ RIP

• **حالت اتصال Link State**

◦ اینترنت اخیر

◦ OSPF

## الگوریتم های مسیریابی در اینترنت :

به طور کلی ما دو دسته اصلی، الگوریتم مسیریابی داریم: (بدون در نظر گرفتن مسیر یابی SDN)

• **(بردار فاصله) Distance Vector**

روش قدیمی تری که در اینترنت استفاده میشده است. در شبکه های ناول و DEC مورد

توجه قرار گرفت.

## • Link State (حالت اتصال)

اخیرا از آن در اینترنت استفاده میشود و جدید تر است مانند الگوریتم مسیر یابی OSPF

# Routing protocols

**Routing protocol goal:** determine “good” paths (equivalently, routes), from sending hosts to receiving host, through network of routers

- path: sequence of routers packets will traverse in going from given initial source host to given final destination host
- “good”: least “cost”, “fastest”, “least congested”
- routing: a “top-10” networking challenge!

هدف یک پروتوكل مسیر یابی: مشخص کردن مسیر های خوب از هاست فرستنده به هاست گیرنده در طول روتر های شبکه است.

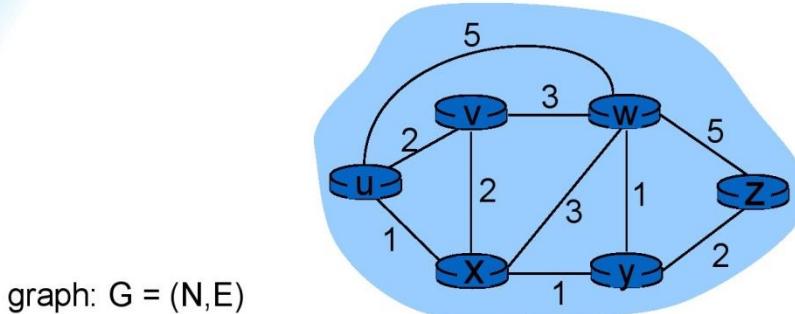
(مسیر): به مسیری گویند که در آن دنباله ای از روتراها قرار دارند که پکتهای دیتا، در طول سفرشان از مبدأ به مقصد، به ترتیب، ازین روترها عبور میکند.

**Good**: گاهی اوقات به معنی کمترین هزینه و سرمایه گذاری جهت ارسال اطلاعات می باشد و گاهی اوقات به معنی سریع تری مسیر جهت ارسال اطلاعات و گاهی اوقات هم مسیری که از دحام کمتری دارد می باشد.

**Routing**: از ده چالش برتر شبکه می باشد و هدف آن انتخاب مسیر مناسب برای ارتباط می باشد.

یک شبکه را میتوان به صورت یک گراف در نظر گرفت که شامل چند گره و لینک های بین گره ها می باشد:

## Graph abstraction of the network



$$N = \text{set of routers} = \{ u, v, w, x, y, z \}$$

$$E = \text{set of links} = \{ (u,v), (u,x), (v,w), (v,x), (w,z), (x,y), (y,z), (x,w), (y,w) \}$$

*aside:* graph abstraction is useful in other network contexts, e.g., P2P, where  $N$  is set of peers and  $E$  is set of TCP connections

9

$N$  مجموعه ای از روتراها و  $E$  مجموعه ای از لینک ها هستند که به کمک این دو میتوانیم شبکه را به صورت مجموعه ای گراف متشکل از راس ها و یال ها نشان دهیم.

در شکل بالا هر گره (که در اینجا همان روترها هستند) با حروف لاتین اسم گذاری گردیده است، لینک های بین این گره ها (که همان Edge یا لبه ها هستند) را مشخص شده است و روی هر لینک آن نوشته شده است. در اینجا Cost هزینه ریالی یا Delay و یا هر پارامتر دیگری که از نظر طراح مهم باشد می تواند در نظر گرفته شود.

برای پاسخ به این سوال که مسیری با حداقل هزینه (Cost) کدام است باید به موارد زیر  
پردازیم»

هر لینک که به صورت  $C(X, X')$  نشان میدهیم. برای مثال در شکل صفحه بعد داریم:

$$C(W, Z) = 5$$

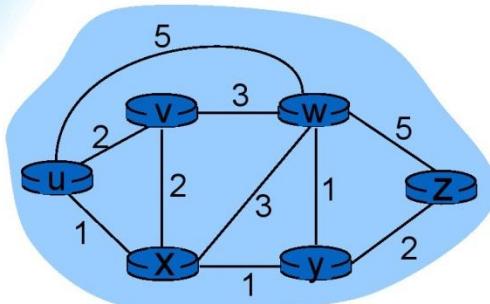
که در واقع همان عدد نوشته شده بین دو گره  $W$  و  $Z$  می باشد.

در این صورت ما اگر Cost هر مسیر را بخواهیم محاسبه کنیم داریم:

$$\text{Cost Of Path} = (X_1, X_2, X_3, \dots, X_{p-1}, X_p) = C(X_1, X_2) + C(X_2, X_3) + \dots + C(X_{p-1}, X_p)$$

و ازین طریق این امکان وجود دارد مسیری با کمترین هزینه در گرافی مشخص با مسیری مشخص رو پیدا کنیم.

## Graph abstraction: costs



$c(x, x')$  = cost of link  $(x, x')$   
e.g.,  $c(w, z) = 5$

cost could always be 1, or  
inversely related to bandwidth,  
or inversely related to  
congestion

$$\text{cost of path } (x_1, x_2, x_3, \dots, x_p) = c(x_1, x_2) + c(x_2, x_3) + \dots + c(x_{p-1}, x_p)$$

**key question:** what is the least-cost path between  $u$  and  $z$ ?  
**routing algorithm:** algorithm that finds that least cost path

برای بررسی بهتر ما میتوانیم الگوریتم های مسیر یابی را دسته بندی کنیم. که این دسته بندی به دو صورت امکان پذیر است:

- دسته بندی از لحاظ اطلاعات (Informatin)
- دسته بندی از نظر پویا و ایستا بودن

دسته بندی بر اساس اطلاعات:

به دو حالت Global و Decentralized تقسیم میشود:

### **:Global**

در این حالت همه مسیریاب ها دسترسی به تمامی اطلاعات توپولوژی ها و هزینه لینک ها را دارند.

مثل الگوریتم Link State

### **:Decentralized**

در این دسته بندی روتر فقط همسایه هایی که به صورت فیزیکی به آنها متصل هستند را می شناسد و فقط هزینه ارتباط با همان همسایه ها را می داند. در چنین حالتی فرایند محاسبه بهترین مسیر یک فرایند Iterative (تکراری و مرحله ای) می باشد. یعنی در ابتدا روتر فکر می کند یک مسیر بهترین مسیر می باشد ولی با گذشت زمان و با جدید شدن اطلاعات Share شده با نود های همسایه متوجه می شود که مسیر بهتری وجود دارد و مرحله به مرحله ممکن است مسیر تغییر بکند.

مثل الگوریتم Distance Vector

# Routing algorithm classification

*Q: global or decentralized information?      Q: static or dynamic?*

*global:*

- all routers have complete topology, link cost info
- “link state” algorithms

*decentralized:*

- router knows physically-connected neighbors, link costs to neighbors
- iterative process of computation, exchange of info with neighbors
- “distance vector” algorithms

*static:*

- routes change slowly over time

*dynamic:*

- routes change more quickly
  - periodic update
  - in response to link cost changes

11

دسته بندی ایستا و پویا:

: (ایستا) **Static**

در این حالت مسیرها در طی زمان به آهستگی تغییر میکند. در واقع چون در اساس آن به صورت اتوماتیک در خصوص مسیر یابی تصمیم گیری نمی شود این اتفاق می افتد. الگوریتم های وابسته به توپولوژی و پارامتر های توپولوژی معمولاً به این صورت می باشند.

: (پویا) **Dynamic**

الگوریتم هایی هستند که در آن مسیر ها خیلی سریع تر تغییر میکند. دارای Periodic Update هستند. این الگوریتم ها معیار های مسیریابی شون هزینه هست مثل تاخیر، این نوع الگوریتم ها دایماً مسیر را تعویض می کنند و بهترین مسیر را در لحظه انتخاب می کنند.

### الگوریتم مسیریابی **Link State**

این الگوریتم مسیر یابی از الگوریتم Dijkstra (دایکسترا) برای محاسبه بهترین مسیر استفاده می کند.

این الگوریتم نیاز دارد تا توپولوژی کل شبکه را به همراه هزینه همه نودها بداند که این کار به کمک Share (به اشتراک گذاری) کردن اطلاعات بین نود ها صورت میگیرد که به این کار **Link State Broadcast** می گویند. پس نتیجه میگیریم در این الگوریتم همه نود ها دارای اطلاعات یکسانی هستند چون یک شبکه واحد می باشد.

وظیفه الگوریتم دایکسترا این است که بهترین مسیر از یک نود به همه نودهای آن شبکه پیوسته Forwarding Table ایجاد می شود که برای کردن پکت هایی که به این نود می رسد تا به نود های دیگر ارسال شود استفاده می شود تا بهترین مسیر انتخاب شود.

# A link-state routing algorithm

## Dijkstra's algorithm

- net topology, link costs known to all nodes
  - accomplished via “link state broadcast”
  - all nodes have same info
- computes least cost paths from one node (‘source’) to all other nodes
  - gives *forwarding table* for that node
- iterative: after  $k$  iterations, know least cost path to  $k$  dest.’s

## notation:

- $c(x,y)$ : link cost from node  $x$  to  $y$ ;  $= \infty$  if not direct neighbors
- $D(v)$ : current value of cost of path from source to dest.  $v$
- $p(v)$ : predecessor node along path from source to  $v$
- $N'$ : set of nodes whose least cost path definitively known

12

برای این الگوریتم دو مراحل اصلی وجود دارد: **Iterative** و **Finalization**.  
مراحل اولیه این الگوریتم را در مراحل اولیه پوشش نمی‌کند.

هزینه لیتک از  $X$  به  $Y$  که اگر  $y$  وجود نداشت بی نهایت می‌شود.

هزینه رسیدن از سورس به مقصد می‌باشد.

هزینه قبلی رسیدن به مقصد نود فعلی به نود مقصد می‌باشد.

نود هایی هستند که کمترین هزینه برای مسیریابی تا آن ها محاسبه شده است.

مشاهده بدنی الگوریتم:

## Dijkstra's algorithm

```
1 Initialization:
2    $N' = \{u\}$ 
3   for all nodes  $v$ 
4     if  $v$  adjacent to  $u$ 
5       then  $D(v) = c(u,v)$ 
6     else  $D(v) = \infty$ 
7
8 Loop
9   find  $w$  not in  $N'$  such that  $D(w)$  is a minimum
10  add  $w$  to  $N'$ 
11  update  $D(v)$  for all  $v$  adjacent to  $w$  and not in  $N'$  :
12     $D(v) = \min(D(v), D(w) + c(w,v))$ 
13  /* new cost to  $v$  is either old cost to  $v$  or known
14  shortest path cost to  $w$  plus cost from  $w$  to  $v$  */
15 until all nodes in  $N'$ 
```

13

در خط اول تا هفتم(فاز مقدار دهی اولیه)  $N'$  خوانده می شود و آن را  $U$  قرار می دهیم سپس برای همه نود های  $V$  در صورتی که در همسایگی  $U$  قرار دارد  $D(V)=C(U.V)$  و در غیر اینصورت  $D(V)$  برابر بی نهایت است.

در خط هشتم تا پانزدهم(فاز اجرا) در هر گام باید یک  $N'$  که در  $D(W)$  نیست باید انتخاب شود که نسبت به بقیه که  $N'$  نیستند حداقل باشد. سپس  $D(W)$  رو به  $N'$  اضافه می کنیم و برای همه همسایه های  $W$  اون رو محاسبه می کنیم:

$$D(V) = \min(D(V), D(W) + C(W, V))$$

این بدین معناست که ما یک نود با هزینه دسترسی کمتر نسبت به بقیه نودها پیدا کرده ایم و می خواهیم ببینیم هزینه ای که از طریق آن نود به مقصد می رویم کمتر خواهد بود یا همان مسیر قبلی که انتخاب کرده ایم. به این صورت به صورت تکرار در حلقه بهترین مسیر مشخص و انتخاب خواهد شد.

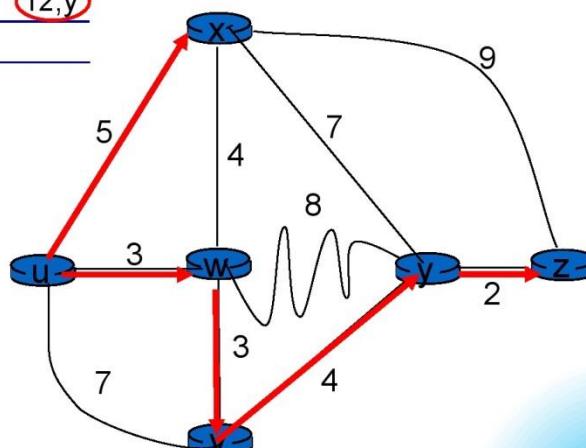
به هدف شناسایی بهترین مسیرها از نود  $U$  به همه نودها اجرای الگوریتم در مثال صفحه بعد قابل مشاهده است:

## Dijkstra's algorithm: example

Step	$N'$	$D(v)$	$D(w)$	$D(x)$	$D(y)$	$D(z)$
		$p(v)$	$p(w)$	$p(x)$	$p(y)$	$p(z)$
0	$u$	7, $u$	3, $u$	5, $u$	$\infty$	$\infty$
1	$uw$	6, $w$		5, $u$	11, $w$	$\infty$
2	$uwx$	6, $w$			11, $w$	14, $x$
3	$uwxv$				10, $v$	14, $x$
4	$uwxvy$					12, $y$
5	$uwxvzy$					

notes:

- ❖ construct shortest path tree by tracing predecessor nodes
- ❖ ties can exist (can be broken arbitrarily)



14

یک جدول می کشیم و در مرحله اول  $N'$  برابر  $U$  می شود و در هر مرحله گره بعدی تا مقصد را اضافه می کنیم.

سپس در هر محله  $D$  آن گره را محسابه و در جدول می گذاریم تا با بقیه اطلاعات جدول مقایسه کنیم.

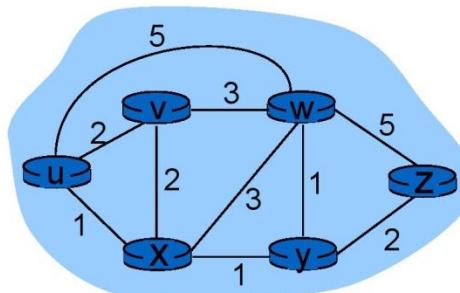
حالا جهت پر کردن جدولی که ستون و ردیف آن را مشخص کردیم در هر گره هزینه مسیر را به همراه گره قبلی در جدول می نویسیم. مثلا در صورت اینکه  $N'$  برابر  $U$  باشد ما با هزینه ۷ و توسط خود  $U$  با آن ارتباط داریم ( $U, 7$ ) و یا در خط بعدی ( $N' = UW$ ) برای رسیدن به  $W$  از مقصد  $U$  و مسیری که  $V$  از آن می گذرد بایستی هزینه  $3+3=6$  یا همان ۶ را داشته باشیم ( $W, 6$ ).

با توجه به اسلاید بالا برای هر کدام کمترین هزینه را انتخاب و مسیر مناسب را میتوانیم مشخص کنیم که در جدول با قرمز دور آن خط کشیده شده است.

مثال دیگری از دایکسترا که باید خودمان آن را بررسی کنیم:

## Dijkstra's algorithm: another example

Step	$N'$	$D(v), p(v)$	$D(w), p(w)$	$D(x), p(x)$	$D(y), p(y)$	$D(z), p(z)$
0	u	2,u	5,u	1,u	$\infty$	$\infty$
1	ux	2,u	4,x		2,x	$\infty$
2	uxy	2,u	3,y			4,y
3	uxyv		3,y			4,y
4	uxyvw					4,y
5	uxyvwz					

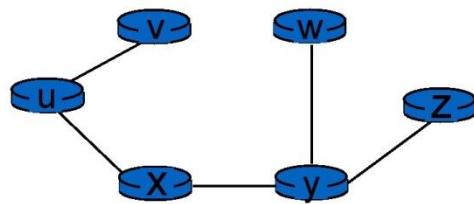


\* Check out the online interactive exercises for more examples: [http://gaia.cs.umass.edu/kurose\\_ross/interactive/](http://gaia.cs.umass.edu/kurose_ross/interactive/)

نمایش Forwarding Table حاصل از بهترین مسیر ها در شکل بالا:

## Dijkstra's algorithm: example (2)

resulting shortest-path tree from u:



resulting forwarding table in u:

destination	link
v	(u,v)
x	(u,x)
y	(u,x)
w	(u,x)
z	(u,x)

16

جهت بررسی پیچیدگی شبکه در صورتی که ما N نود در شبکه داشته باشیم در هر ما همه نود های W را که در N نیستند باید رد نظر بگیریم و در هر مرحله یکی از آن ها کم می شود پس:

$$\text{تعداد محاسبات} = \frac{N(N+1)}{2}$$

پس در صورتی که ما یک نود به نودهایمان اضافه کنیم پیچیدگی این محاسبات به توان دو می شود( $O(N^2)$ ).

در صورتی که بخواهیم آن را بهتر محاسبه کنیم پیچیدگی آن را می توانیم به صورت  $O(N \log n)$  محاسبه کنیم.

**Oscillattion**: یا همان جابجایی بی نهاست مسیر می باشد که در این الگوریتم این مسدله امکان دارد. بخصوص اگر معیار مسیر یابی ما ترافیک روی لینک ها باشد این اتفاق حتما می افند در واقع بنا به ترافیک هر مسیر دایما مسیر ارسال به مقصد تغییر میکند.

مانند مثال صفحه بعدی که در آن Link Cost Carried Traffic (هزینه لینک ها برابر با ترافیک حمل کننده آن) می باشد.

همانطور که در شکل قابل مشاهده است (شکل سمت چپ) در لحظه اول از لینک های B,C,D از طریق نود های همسایه A و D به مقصد B قابل استفاده است ولی در لحظه بعدی این سه نود فقط از طریق همسایگی نود D می توانند به مقصد A بروند و در تصویر بعدی از طریق همسایگی نود B و سپس مجددا از طریق همسایگی نود D (شکل سمت راست) که نشان می دهد در هر لحظه بسته به حجم ترافیک مسیریابی تا مقصد تغییر میکند و بهترین مسیر انتخاب می گردد.

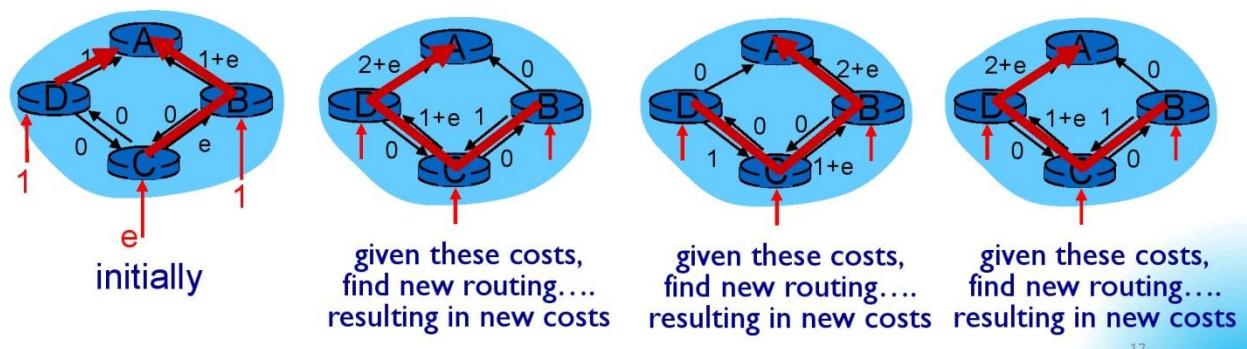
# Dijkstra's algorithm, discussion

*algorithm complexity:* n nodes

- each iteration: need to check all nodes, w, not in N
- $n(n+1)/2$  comparisons:  $O(n^2)$
- more efficient implementations possible:  $O(n \log n)$

*oscillations possible:*

- e.g., support link cost equals amount of carried traffic:



17

## الگوریتم Distance Vector

این الگوریتم از یک تساوی به نام تساوی بلمن فورد (Bellman-Ford Equation) استفاده می‌کند.

در این تساوی اگر  $D_x(Y)$  برابر مسیری با کمترین هزینه از X به Y باشد به ازای همه همسایه‌های X به نام V

$$D_x(Y) = \text{Min} \{C(X,V) + D_v(Y)\}$$

فرض کنید یک نودی (مثل نود X) داری سه همسایه می‌باشد که میخواهیم ما مسیری با کمترین هزینه به نود Y را بدست آوریم. پس باید بررسی کنیم که از مسیر همسایه اول چقدر

هزینه کمتر داریم به اضافه کمترین هزینه رسیدن همسایه اول به مقصد، سپس بررسی کمترین هزینه همسایه دوم به اضافه کمترین هزینه رسیدن از همسایه دوم به مقصد و در نهایت بررسی کمترین هزینه همسایه سوم به همراه کمترین هزینه رسیدن از همسایه سوم به مقصد.

## Distance vector algorithm

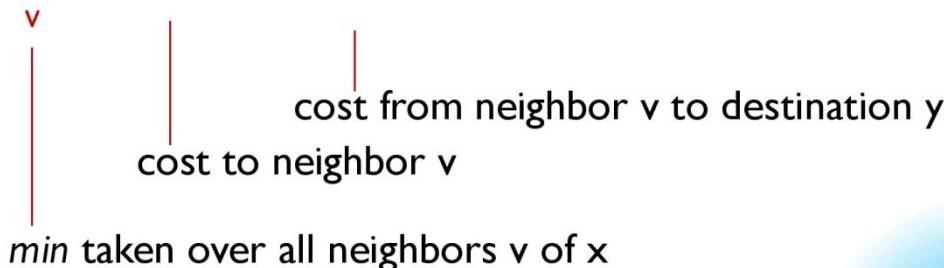
*Bellman-Ford equation (dynamic programming)*

let

$d_x(y) := \text{cost of least-cost path from } x \text{ to } y$

then

$$d_x(y) = \min \{ c(x, v) + d_v(y) \}$$



4 of 31

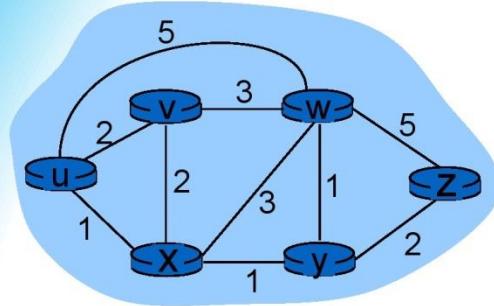
در مثال صفحه بعد

یا همان کمترین هزینه از مبدا  $V$  به مقصد  $Z$  (که سه حالت را ممکن دارد) حالتی است که از  $X$  به  $Y$  و از  $Y$  به  $Z$  برود که هزینه آن ۵ می شود. بنا به توضیح داده شده میتوان  $D_w(Z)$  را ۳ و  $D_x(Z)$  را ۴ برابر گرفت.

بر اساس سه نتیجه بدست آمده بالا  $D_u(Z)$  برابر ۴ می شود:

$$D_u(Z) = \text{Min} \{ C(U,V) + D_v(Z), C(U,X) + D_x(Z), C(U,W) + D_w(Z) \} = \\ \text{Min} \{ 2+5, 1+3, 5+3 \} = \text{Min} \{ 7, 4, 8 \} = 4$$

## Bellman-Ford example



clearly,  $d_v(z) = 5$ ,  $d_x(z) = 3$ ,  $d_w(z) = 3$

B-F equation says:

$$\begin{aligned} d_u(z) &= \min \{ c(u,v) + d_v(z), \\ &\quad c(u,x) + d_x(z), \\ &\quad c(u,w) + d_w(z) \} \\ &= \min \{ 2 + 5, \\ &\quad 1 + 3, \\ &\quad 5 + 3 \} = 4 \end{aligned}$$

node achieving minimum is next hop in shortest path, used in forwarding table

5

با توجه به محاسبات صورت گرفته نودی که کمترین مسیر با کمترین هزینه رو در اینجا به ما بین همسایه ها می دهد Next Hop ما می شود و باید آن را در جدول مسیریابی قرار دهیم. البته تمام این محاسبات را خود الگوریتم برای ما انجام می دهد.

## Distance vector algorithm

- $D_x(y)$  = estimate of least cost from  $x$  to  $y$ 
  - $x$  maintains distance vector  $D_x = [D_x(y): y \in N]$
- node  $x$ :
  - knows cost to each neighbor  $v$ :  $c(x,v)$
  - maintains its neighbors' distance vectors. For each neighbor  $v$ ,  $x$  maintains  $D_v = [D_v(y): y \in N]$

تقریبی از کمترین مسیر از  $X$  به  $Y$  می باشد. علت تقریبی در نظر گرفتن آن احتمال وجود مسیرهایی می باشد که هنوز کشف نشده است.

همیشه خود  $X$  یک Vector به اسم  $D_x$  برای خودش نگه میدارد که این  $D_x$  برابر است (برای همه  $Y$  های عضو  $N$  در شبکه).

$$D_x = [D_x(Y): Y \in N]$$

که این یعنی می بایست برای همه نودهای شبکه، Distance از خودمان تا آن نودها را نگهداری کنیم.

این وکتور بدست آمده را نود به همه همسایه‌های خود به اشتراک می گذارد. در واقع در نودها فقط می تواند وکتور همسایه‌های خودشان را ببینند. در نتیجه هر نود فقط Distance Vector خودش را به همسایه‌ها می دهد و نه برای بقیه را ولی Distance Vector خود را پیوسته ازین طریق به روز میکند.

پس در نتیجه نود  $X$  اطلاعات زیر را دارد:

- هزینه به هر کدام از همسایه‌های خودش

## Distance vector algorithm

*key idea:*

- from time-to-time, each node sends its own distance vector estimate to neighbors
- when  $x$  receives new DV estimate from neighbor, it updates its own DV using B-F equation:

$$D_x(y) \leftarrow \min_v \{c(x,v) + D_v(y)\} \text{ for each node } y \in N$$

- ❖ under minor, natural conditions, the estimate  $D_x(y)$  converge to the actual least cost  $d_x(y)$

در الگوریتم Distance Vector به صورت دوره ای هر نод خودش را به همسایه های خود می دهد. وقتی که یک Distance Vector  $X$  جدید(متفاوت با قبلی) دریافت کرد Distance Vector خودش را به کمک الگوریتم بلمن فورد تغییر می دهد و در صورت وجود تغییر آن را به همسایه های خودش معرفی می کند.

در شرایط عادی Estimate  $D_x(Y)$  بعد از چند جابجایی روی کمترین هزینه  $D_x(Y)$  تغییر می دهد. اما گاهی اوقات این فرایند ممکن است خیلی هم زمان بر باشد.

# Distance vector algorithm

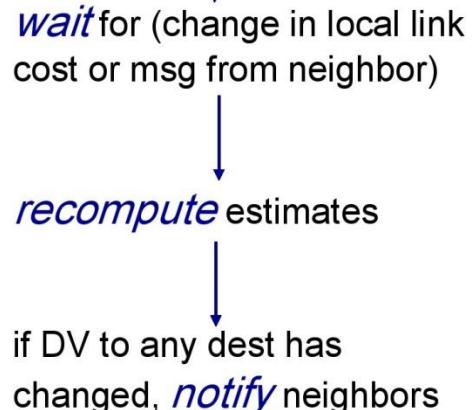
*iterative, asynchronous:* each local iteration caused by:

- local link cost change
- DV update message from neighbor

*distributed:*

- each node notifies neighbors *only* when its DV changes
  - neighbors then notify their neighbors if necessary

*each node:*



8

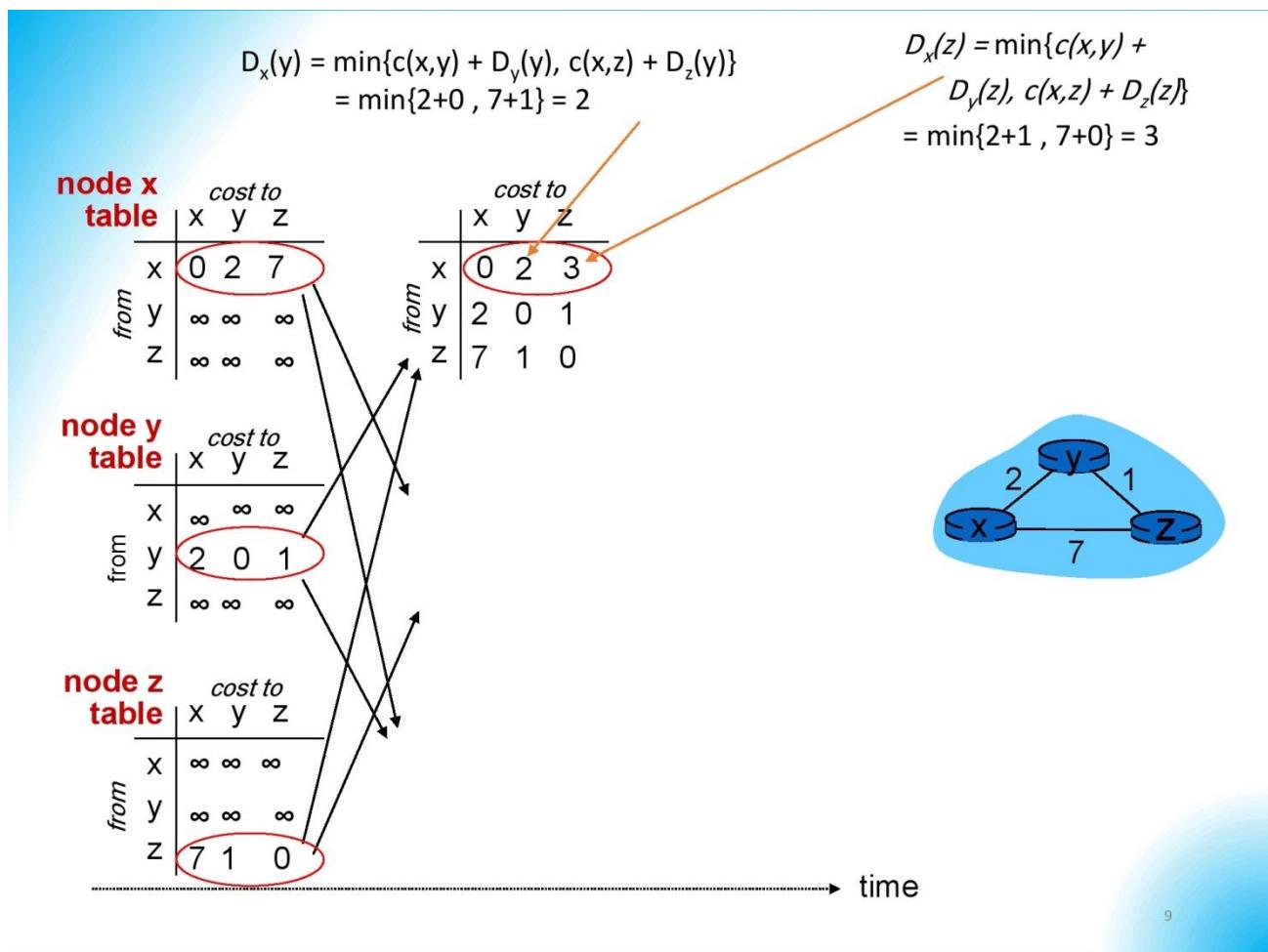
## نحوه تکرار (Iterative) در الگوریتم Distance Vector

زمانی رخ می دهد که هزینه لینک محلی تغییر کند و یا پیام به روز رسانی همسایه ها به دستش برسد. پس زمانی Distributed Distance Vector انجام می شود که آن تغییر کرده است.

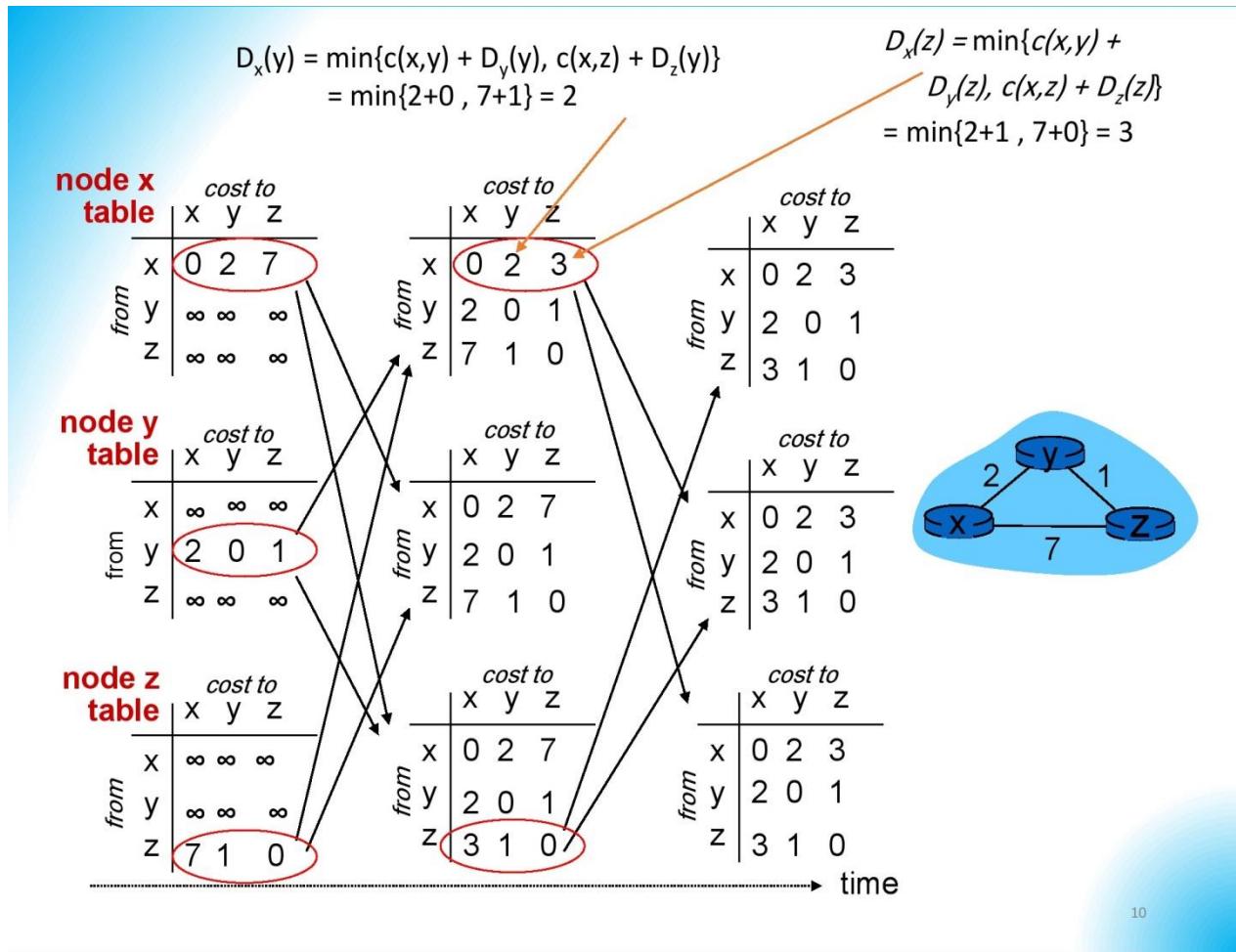
حلقه (Loop) موجود رد شکل بالا نمایانگر این تکرار و شرایط آن برای هر نод می باشد. در این شکل نod پیوسته در انتظار پیغام تغییر Distance Vector توسط همسایه های خود می باشد. به محض تغییرات توسط همسایه ها گزارش شد محسابه مجدد تقریب ها (Recompute Estimates) صورت میگیرد تا بهترین مسیر های جدول Distance Vector برای خود نod به

روز رسانی شود و اگر در نهایت جدول خود تود تغییر کرد این تغییرات به سایر نود های همسایه اطلاع رسانی می شود تا آن ها نیز تغییرات لازم را لحاظ نمایند.

در مثال صفحه بعد یک شبکه به همراه Link Cost های آن مشخص شده است و در شروع کار(لحظه اول) نود X فقط هزینه ارتباطات مستقیم خود را می داند یعنی ارتباط مستقیم X با X,Y,Z و در جدول هزینه خود سایر ارتباطات را چون ندارد بی نهایت تلقی می کند و همین طور این شرایط برای نود های Y و Z به همین صورت است با در اختیار گذاشتن جدول Destance Vector خود به سایر نود های همسایه در مرحله بعدی می بینیم که جدول هزینه بروز رسانی شده و همه ردیف های خود را به کمک اطلاعات هزینه همسایه ها پر میکند و هر نود یک جدول کامل که راهنمای مسیریاب برای کمترین هزینه تا مقصد ها می باشد را بدست می آورد.



در مرحله بعدی مجددا با داشتن اطلاعات جدید جدول همه نود ها توسط خود نودها بروزرسانی می شود و برای هر نود بهترین هزینه جدول مشخص و بکارگیری می شود. سپس جهت تصمیم گیری مجدد به بقیه نود ها جدول خود را اطلاع می دهد.



و در نهایت نود ها به مرحله پایداری می رساند چون تغییر دیگری در جدول ها صورت نمی گیرد تا اتفاق جدیدی در شبکه رخ دهد و نود ها از جدول نهایی بدست آمده جهت مسیریابی با کمترین هزینه استفاده می کنند.

الگوریتم Count To Infinity یک سری ایرادها دارد. یکی از آنها به نام Distance Vector که در جلسه بعدی مورد بررسی قرار خواهد گرفت.

## جلسه ششم:

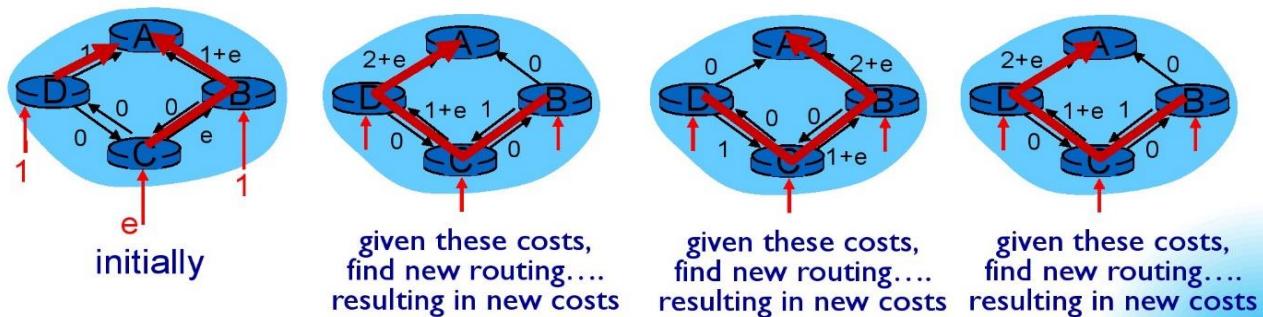
### Dijkstra's algorithm, discussion

*algorithm complexity:* n nodes

- each iteration: need to check all nodes, w, not in N
- $n(n+1)/2$  comparisons:  $O(n^2)$
- more efficient implementations possible:  $O(n \log n)$

*oscillations possible:*

- e.g., support link cost equals amount of carried traffic:



17

این شکل یکی از اتفاقات نامطلوب در مسیریابی را توضیح میدهد.

در صورتی که معیار در مسیر یابی Link State میزان ترافیکی باشد که روی لینکها وجود دارد،

امکان اینکه در مسیرها Oscillation رخ بدهد وجود دارد و به عبارتی ممکن است مسیرها

بطور متناوب و پی در پی تغییر کنند.

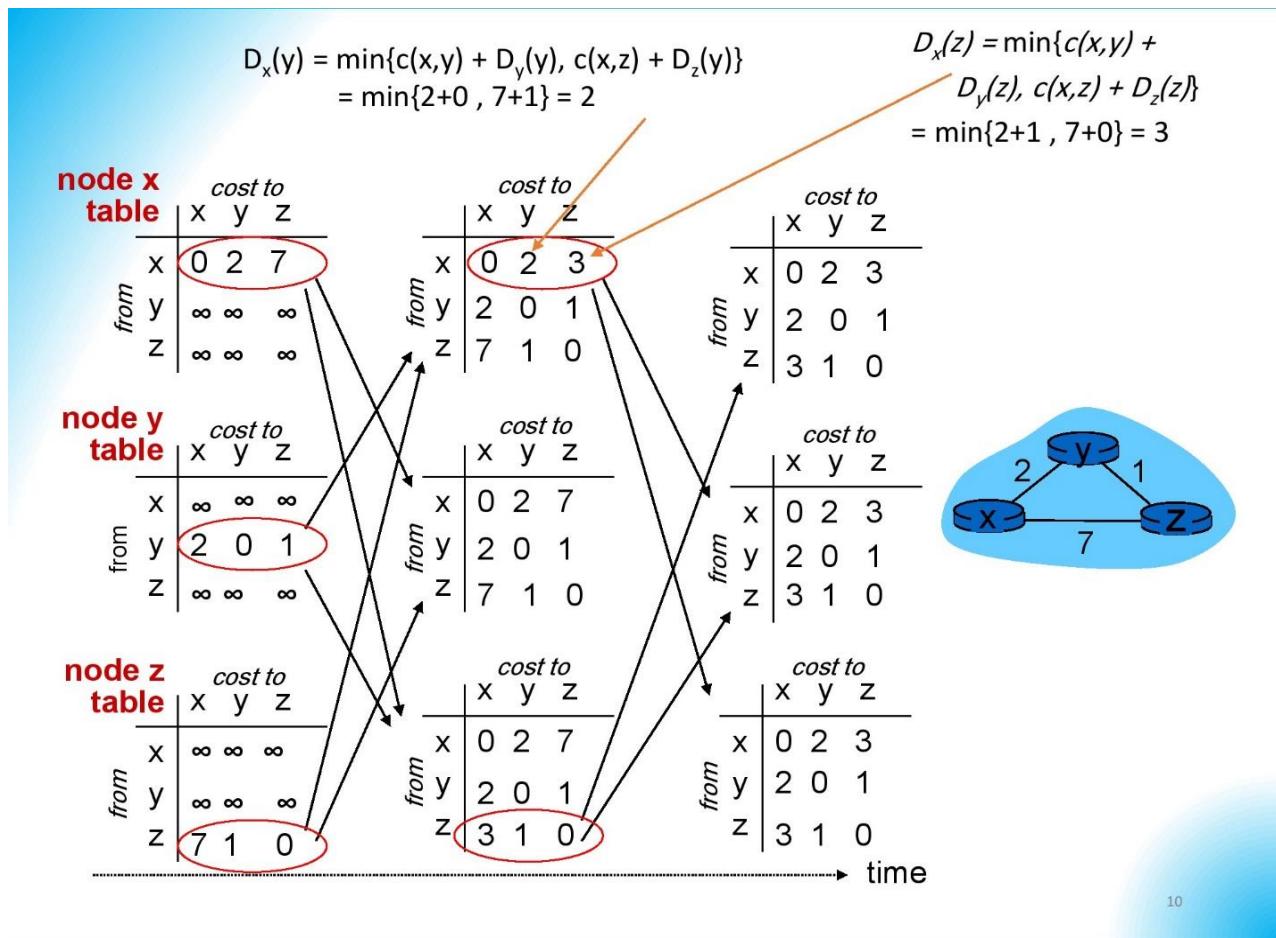
در این مثال گره B و D ترافیکی به حجم 1 و گره C ترافیک کوچکی به حجم E را ، برای گره

A ارسال می کنند. در شروع کار گره های B و D مسیر مستقیمی با A دارند ، این مسیر را برای

عبور اطلاعات انتخاب میکنند. گره C چون این دو مسیر برایش یکسان است یکی را (مثلاً B) را انتخاب میکند.

پس از اجرا مجدد الگوریتم Dijkstra نودهای B و C مشاهده می کنند که از طریق مسیر نod D ترافیک کمتری وجود دارد لذا مسیر خود را عوض کرده و از طریق نod D ترافیک خود را عبور می دهند .

در بروز رسانی سوم نودهای B و C و D درمی یابند که از مسیر B به A ترافیک کمتری (در حد صفر) وجود دارد ، لذا همگی مسیر خود را به سمت B عوض می کنند و این اتفاق بصورت مکرر اتفاق می افتد . که این موضوع اتفاق ناگواری در مسیریابی است.



10

روش مسیر یابی Distance Vector از تساوی Bellman-Ford استفاده می کند . در این

روش هر نод جدول خود را بروز رسانی کرده و سپس آنرا در صورتی که تغییر کرده باشد به

همسایه های مجاور خود اعلام می کند ( Updated Distance Vector )

در گام اول هر نود فقط فاصله از همسایه های خود را دارد و فاصله از سایر نودها را برابر بینهایت

می گذارد. در گامهای بعدی هر نود با توجه به  $Dv$  های ( Distance Vector ) دریافتی از

همسایه هایش جدول خود را بروز کرده و در صورتی که در جدولش تغییری رخ داده باشد ، آنرا

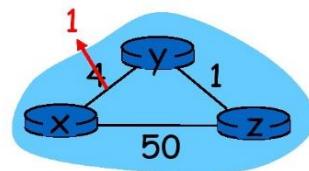
برای همسایه هایش ارسال می کند و اگر تغییری نکرده باشد لذا ارسالی هم رخ نخواهد داد.

در مثال بالا پس از سه مرحله ، تمام نودها ، مسیرهای رسیدن به سایر نودها را در خودشان Update شده دارند و همچنین فاصله رسیدن به نod مقصد خود را هم دارند. مثلاً nod X می‌داند که فاصله اش تا nod Z برابر سه است و باید اینکار را از مسیر Y انجام دهد و یا اینکه فاصله نودهای X و Y برابر ۲ است

## Distance vector: link cost changes

### *link cost changes:*

- ❖ node detects local link cost change
- ❖ updates routing info, recalculates distance vector
- ❖ if DV changes, notify neighbors



**“good news travels fast”**

$t_0$ : y detects link-cost change, updates its DV, informs its neighbors.

$t_1$ : z receives update from y, updates its table, computes new least cost to x, sends its neighbors its DV.

$t_2$ : y receives z's update, updates its distance table. y's least costs do *not* change, so y does *not* send a message to z.

\* Check out the online interactive exercises for more examples: [http://gaia.cs.umass.edu/kurose\\_ross/interactive/](http://gaia.cs.umass.edu/kurose_ross/interactive/)

11

زمانی که هزینه لینکها (Link Cost) تغییر می‌کند ، چه اتفاقی رخ می‌دهد؟؟

در این مثال Cost بین X و Y از چهار به یک تغییر کرده است (Good News). در این حالت نودهای X و Y متوجه این تغییر می‌شوند و تغییر Cost خود را تشخیص می‌دهند ، لذا

جداول خود را بروز می کنند و Vector جدید تغییر یافته خود را به همسایگانشان از جمله  $Z$  اطلاع می دهند. مثلا  $Y$  به اعلام میکند: " من از  $X$  به  $Y$  یک مسیر با Cost یک دارم ". پس از دریافت این پیام توسط  $Z$  ، از آنجاییکه قبل از  $X$  یک مسیر با هزینه  $4+1=5$  داشته و الان یک پیشنهاد با هزینه  $1+1=2$  دارد که از قبلی کمتر است لذا آنرا می پذیرد و جدول خود را بروز می کند و این را برای همسایه های خود اعلام می کند و پس از این جداول به حالت پایدار می رود. زمانی که هزینه یک لینک کم می شود به عنوان یک خبر خوب ( Good News ) به سرعت در شبکه پخش شده و شبکه پایدار می شود. ( در این مثال در سه مرحله شبکه پایدار می شود) . اگر بخواهیم این مثال را بصورت مرحله مرحله بیان نماییم:

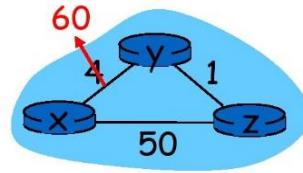
- ۱- در گام اول یعنی لحظه  $T_0$  ، نود  $Y$  تغییر Cost خود را تشخیص می دهد و  $Dv$  خود را بروز رسانی کرده و برای همسایه هایش ارسال می کند
- ۲- در گام دوم یعنی لحظه  $T_1$  نود  $Z$  این بسته را دریافت کرده و پس از انجام محاسبات جدول خود را تغییر می دهد و جدول بروز شده را برای همسایه هایش ارسال می کند
- ۳- در گام سوم یعنی لحظه  $T_2$  ، نود  $Y$  ،  $Dt$  ارسال شده از  $Z$  را دریافت کرده ولی چون مسیر خودش تا نود  $X$  کوتاهتر و بهتر است ، تغییری در جدولش نمی دهد و جدولی هم برای سایر همسایه ها ارسال نمی کند.

در واقع با دو تغییر در  $Dv$  شبکه به حالت نهایی و پایدار می رسد.

## Distance vector: link cost changes

### *link cost changes:*

- ❖ node detects local link cost change
- ❖ *bad news travels slow* - “count to infinity” problem!
- ❖ 44 iterations before algorithm stabilizes: see text



### *poisoned reverse:*

- ❖ If Z routes through Y to get to X :
  - Z tells Y its (Z's) distance to X is infinite (so Y won't route to X via Z)
- ❖ will this completely solve count to infinity problem?

12

در این مثال هزینه لینک بین X و Y از ۴ به ۶۰ افزایش می یابد یعنی خبر بد (Bad News).

حال ببینیم این خبر در جداول چگونه اثر می گذارد؟

قبل از تغییر، هزینه (Cost) از X به Z از مسیر Y برابر ۵ بوده است و هزینه X و Y برابر ۴ بوده و هزینه لینک Y و Z ۱ بوده است.

پس از تغییر، Y می بیند هزینه رسیدن به X از لینک مستقیم برابر ۶۰ است ولی در Dv مربوط به Z هزینه رسیدن به X برابر  $5+4=9$  (حالت قبل از تغییر) است پس با فکر اینکه از

طريق Z می تواند به X دسترسی پیدا کند هزینه خود را  $1+5=6$  می کند و آنرا به همسایگان از جمله Z اعلام می نماید.

این پیام پس از دریافت توسط Z بررسی شده و می بیند هزینه مستقیمش به X برابر ۵۰ است و از طريق Y این هزینه برابر  $6+1=7$  است. نود Z پس از اصلاح جدول DV خود آنرا برای همسایگانش Broadcast می کند. و این سیکل بطور متناوب بین Z و Y تکرار می شود تا زمانی که Z متوجه شود از مسیر مستقیم هزینه کمتری برای رسیدن به X دارد و این هزینه ۵۰ است . حال ۵۰ را در جدول خود بروز کرده و برای Y ارسال می کند در اینجا هم Y جدول خود را به  $1+50=51$  تغییر می دهد و برای Z می فرستد . در چون این عدد  $(1+51=52)$  از مسیر خودش بیشتر است لذا به جدول خود دست نمی زند و کار پایان می پذیرد.

در جدول مربوط به نود Y از اولین آپدیت یعنی ۶ شروع شده و تا ۵۱ یکی بکی اضافه می شود . و در جدول مسیر یابی نود Z هم تغییرات از ۷ شروع شده و تا ۵۰ ادامه می یابد یعنی ۴۴ گام تغییر و تکرار ( Heration ). لذا اخبار بد به کندی در شبکه حرکت می کنند و شبکه خیلی دیر به حالت پایدار خود می رسد. علت اصلی این اتفاق اینست که نودها نمی دانند که این مسیر با این هزینه از طريق کدام نود بدست آمده است. به این ایراد و مشکل Count To Infinity

**Problem** یا مشکل شمارش تا بینهايت می گویند . یک راه حل این مشکل روش Poisoned Reverse است . یعنی زمانی که می بینیم مثلا Z به X از طريق Y ارتباط دارد باید به Y اعلام کند که فاصله من تا X برابر بی نهايت است ( چون از طريق Y به X ارتباط دارد

به  $Y$  اعلام می کند که "هزینه  $Z$  تا  $X$  برابر بینهایت است".) یعنی به جای اینکه  $Z$  به  $Y$  اعلام کند که فاصله اش تا  $X$  برابر ۵ است به  $Y$  می گوید فاصله من تا  $X$  بینهایت است. (البته فقط به  $Y$  این مطلب را می گوید و به سایر همسایه ها هزینه واقعی را می گوید).

اگر در شبکه اتفاق بدی بیفتد و ما از روش **Poisoned Reverse** استفاده کنیم روال به شرح زیر می شود:  $Y$  می بیند که مسیرش به  $X$  برابر ۶۰ شده و می بیند فاصله  $Z$  تا  $X$  بی نهایت است پس جدول خود را به ۶۰ تغییر می دهد و آنرا برای همسایگانش از جمله  $Z$  اعلام می کند. در  $Z$  می بیند که ارتباط مستقیمش تا  $X$  برابر ۵۰ است پس  $Dv$  خود را با  $50$  بروز رسانی کرده و آنرا برای  $Y$  ارسال می کند. در  $Y$  هم هزینه جدید را محاسبه می کند که برابر  $51 = 1 + 50$  است که از ۶۰ کمتر است پس جدول خود را بروز رسانی کرده و  $Dv$  جدید را برای  $Z$  ارسال می کند و از آنجاییکه جداول دیگر تغییری نمی کنند، شبکه پایدار می شود و به حالت تعادل می رسد.

آیا متدهای Count To Infinity مشکل Poisoned Reverse را بطور کامل حل می کند؟؟

# Comparison of LS and DV algorithms

## message complexity

- **LS:** with  $n$  nodes,  $E$  links,  $O(nE)$  msgs sent
- **DV:** exchange between neighbors only
  - convergence time varies

## speed of convergence

- **LS:**  $O(n^2)$  algorithm requires  $O(nE)$  msgs
  - may have oscillations
- **DV:** convergence time varies
  - may be routing loops
  - count-to-infinity problem

## robustness: what happens if router malfunctions?

### LS:

- node can advertise incorrect **link** cost
- each node computes only its own table

### DV:

- DV node can advertise incorrect **path** cost
- each node's table used by others
  - error propagate thru network

13

## مقایسه عملکرد Link State و Distance Vector

پیچیدگی پیام: در LS اگر  $N$  نود داشته باشیم و  $E$  لینک ، تعداد پیامهای رد و بدل شده از

درجه ( $O(Ne)$ ) است زیرا هر نود باید اطلاعات همه نودها و توپولوژی کل شبکه را داشته باشد.

پس با اضافه کردن هر نود یا لینک پیچیدگی زیاد می شود . در DV ارتباط یا تبادل اطلاعات با

همسایه ها صورت می پذیرد و به سایر قسمتهای شبکه ارتباطی ندارد ، لذا اگر تعداد نودها یا

لینکها تغییر کند، سایر پیامها تغییر زیادی نمی کنند فقط در همسایگی ها تغییر ایجاد می

شود. البته مشکل DV زمان همگرایی و سرعت متغیر آن است.

**سرعت همگرایی** : که در LS سرعت همگرایی  $O(N^{82})$  که نیاز به رد و بدل کردن (Ne) پیام دارد ، که ممکن است باعث یک سیکل تکراری و Oscillation هم بشود. در Dv زمان Count To Infinity Loop و یا همگرایی وابسته به شرایط است و ممکن است مشکل Routing Loop ایجاد گردد.

**استحکام بخشی شبکه ( Robustness )** : یعنی اگر نود یا مسیریابی دچار مشکل شود برای شبکه چه مشکلی رخ می دهد ؟

در LS ممکن است هزینه اشتباه در شبکه منتشر ( Advertise ) شود و چون هر نود جدول خود را دارد ، ممکن است فقط قسمت خاصی از شبکه در مسیریابی مشکل پیدا کند و در مسیریابی سایر قسمتهای شبکه اثری بدی نگذارد. لذا اگر روتربی اطلاعات اشتباه داد ، فقط بخشی از شبکه مختل می شود.

در Dv چون Vector یک روتربی توسط همسایه ها مورد استفاده قرار می گیرد و این جدول به همسایه ها ارسال شده و روی آنها تاثیر می گذارد ، پس ممکن است این خطاكل شبکه را پوشش دهد. پس LS در Robustness از Dv بهتر است.

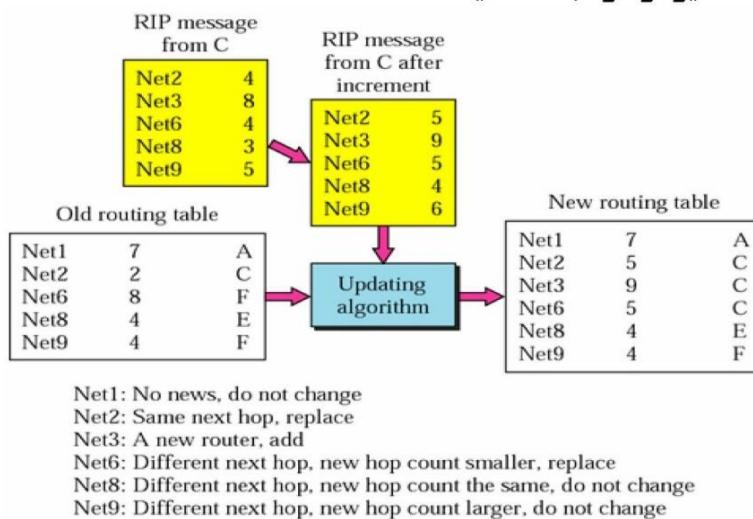
## الگوریتم های مسیریابی در اینترنت

### • بردار فاصله Vector

### • پروتکل RIP

#### • بردار تاخیر : تاخیر از گره به همسایه ها

#### • محاسبه



14 of 31

الگوریتم مسیریابی مورد استفاده در اینتر نت در درون AS ها مبتنی بر بردار فاصله (DV) است

که از پروتکل Rip استفاده می شود. در اینجا فاصله مساوی تاخیر رسیدن یک بسته از یک گره به

گره همسایه است. این معیار مسیر یابی است. مثال: اگر در نود جدول مسیر یابی Old Routing

باشد که فاصله از Net های مختلف و Next Hop آنها در این جدول مشخص شده

است. مثلا در این روتر فاصله رسیدن به Net 1 برابر 7 است و از طریق نود A صورت می پذیرد.

فرض کنید یک پیام از نود C دریافت می شود و در این پیام فاصله نود C تاشبکه های مختلف

مشخص شده است. تاخیر نود A تا نود C 1 واحد است. پس به فاصله های موجود در جدول C

یکی اضافه می شود. حال اعداد رسیده از جدول C را با جدول نود خودمان مقایسه می کنیم. در اینصورت چند حالت رخ می دهد.

در  $Net^2$  قبل از طریق C بوده و چون در حال حاضر هم پیام را از نود C دریافت کرده ، لذا مقدار آن ۵ می شود (حتی با وجود اینکه بیشتر هم شده است).

اطلاعات  $Net^3$  چون وجود نداشته ، به جدول اضافه می شود.

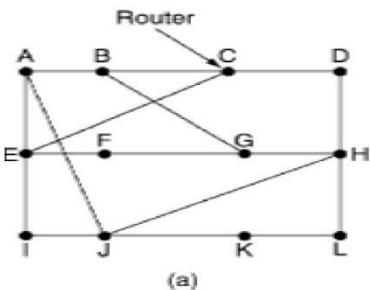
اطلاعات  $Net^6$  چون از مسیر C کمتر است ( $5 < 8$ ). پس فاصله ۵ و Next Hop=C می شود.

در  $Net^8$  و  $Net^9$  چون فاصله کم نشده ، پس تغییری در جدول رخ نمی دهد.

# الگوریتم های مسیریابی در اینترنت

## • بردار فاصله Distance Vector

## • پروتکل RIP



New estimated delay from J

To	A	I	H	K	Line
A	0	24	20	21	8 A
B	12	36	31	28	20 A
C	25	18	19	36	28 I
D	40	27	8	24	20 H
E	14	7	30	22	17 I
F	23	20	19	40	30 I
G	18	31	6	31	18 H
H	17	20	0	19	12 H
I	21	0	14	22	10 I
J	9	11	7	10	0 -
K	24	22	22	0	6 K
L	29	33	9	9	15 K

Vectors received from J's four neighbors

	JA delay is 8	JI delay is 10	JH delay is 12	JK delay is 6
--	---------------	----------------	----------------	---------------

(b)

15 of 31

در این شکل جدول مسیر یابی J محاسبه می گردد . DV تمام همسایه های J یعنی A,I,H,K دریافت گردیده است. هدف محاسبه جدول مسیریابی J است که در آن مسافت تاتمامی نودهای شبکه در آن وجود داشته باشد. در جدول مسیر یابی ، مقصد ( Destination ) ، فاصله و Next Hop درج می گردد

روش محاسبه به این شکل است که تاخیر همسایه با مقدار نود جمع شده و کوچکترین مسیر و همینطور نام آن همسایه ، در جدول درج می گردد.

مثال:

یکی از همسایه های J است که در آن  $D \text{ Cost} = 8 + 40 = 48$  است A

یکی از همسایه های J است که در آن  $D \text{ Cost} = 10 + 27 = 37$  است. I

یکی از همسایه های J است که در آن  $D \text{ Cost} = 12 + 8 = 20$  است H

یکی از همسایه های J است که در آن  $D \text{ Cost} = 24 + 6$  است و از آنجاییکه از مسیر K

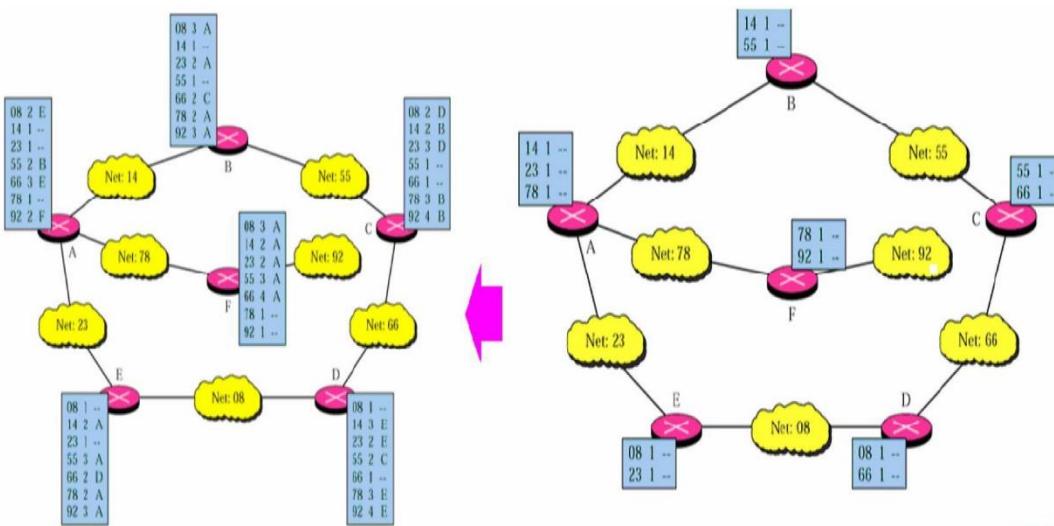
کمترین Cost را داریم پس در جدول مسیر یابی A سطر : Dest=D , Cost=20 , Next :

Hop=20 درج می گردد.

# الگوریتم های مسیریابی در اینترنت

## • بردار فاصله Distance Vector

## • پروتکل RIP



در شبکه سمت راست ، جداول مسیریابی ناقص است و هر روتر فقط اطلاعات شبکه هایی که مستقیماً به آن متصل است را دارد. ولی با چند بار رد و بدل کردن اطلاعات ( Exchange ) ، این جداول تکمیل می گردند.

برای مثال روتر B از روترهای A و C اطلاعات و DV آنها را دریافت کرده و جدول خود را بروز رسانی می کند. ( در این مثال فاصله همه لینکها یک فرض شده است ) . فاصله نود B از Net<sup>66</sup> را برابر ۲ و از طریق Next Hop=C می باشد . و یا رسیدن نود B به شبکه Net<sup>92</sup> از طریق روتر A و با هزینه ۳ می باشد ( یعنی : فاصله نود F تا Net<sup>92</sup> + فاصله نود F تا A + فاصله نود A تا

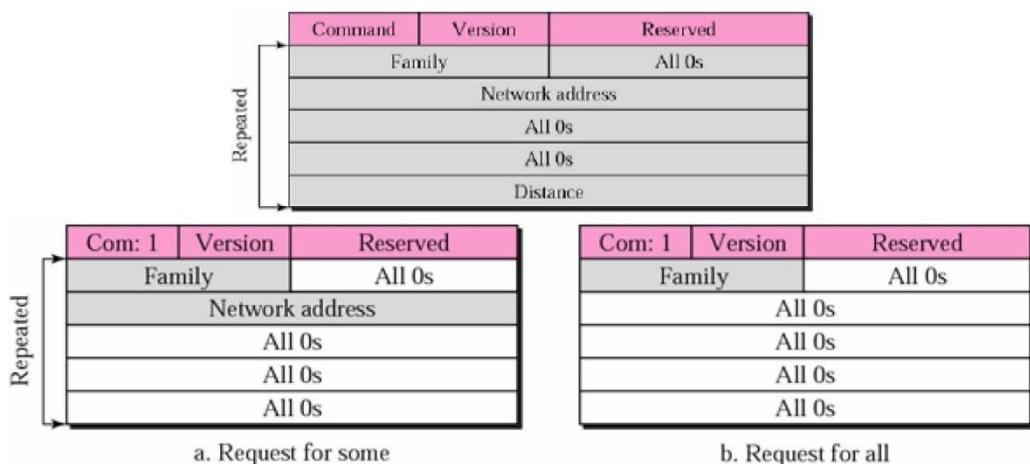
$B = 1+1+1+1 = 4$  و پس از چند بار رد و بدل شدن اطلاعات و DV ها ، همه نودها به هم دسترسی پیدا می کنند.

## الگوریتم های مسیریابی در اینترنت

### • بردار فاصله Vector

### • پروتکل RIP

#### • فرمت بسته ها



17 of 31

در این شکل فرمت بسته های RIP که برای رد و بدل کردن اطلاعات بکار می رود ، نشان داده شده است. در جدول بالا نوع بسته را مشخص می کند. 'S' فیلدی است که تمامی بیت‌هایش صفر است. Network Address و فاصله تا آن Network Address با مشخص می‌شود. این بسته برای همسایه ها ارسال می گردد که بسته های RIP Message هستند.

در جدول بالا قسمت خاکستری ، بسته به اینکه چند سطر در Dv داریم ، این قسمت تکرار می شود (Record قسمت

در جدول پایین سمت چپ (بسته Request )، یک نod از همسایه های خودش برای یک آدرس مشخص ، درخواست اطلاعات می کند و مسیر همسایه را برای رسیدن به یک شبکه بخصوص را می پرسد . حال اگر خواستیم مثلا سه آدرس شبکه را استعلام کنیم ، قسمت Repeat را سه بار تکرار می کنیم و سه درخواست را در یک بسته ارسال می کنیم.

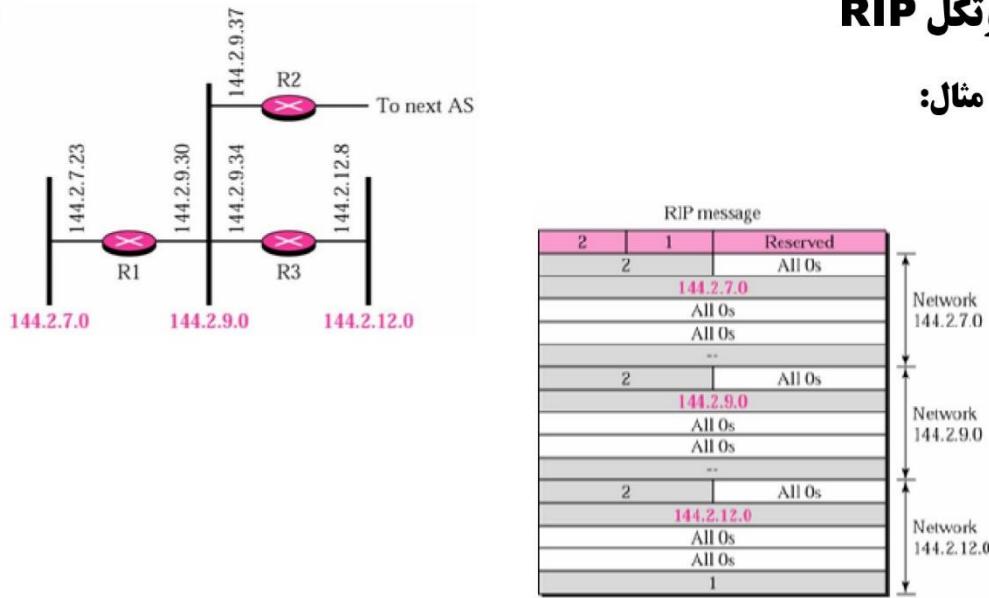
جدول پایین سمت راست مربوط به زمانی است که ما بخواهیم کل جدول را از همسایه ها استعلام بگیریم . در اینجا Network Address خاصی مد نظر نیست و نیاز به تمامی اطلاعات همسایه ها را داریم لذا این فیلد را با ۰ پر می کنیم.

# الگوریتم های مسیریابی در اینترن

## • بردار فاصله Distance Vector

## • پروتکل RIP

• مثال:



18 of 31

در این شکل هدف ارسال Rip Message به سایر نود هاست.

دو فیلد اول Command=۲ و Version=۱ Family=۲ است . سطر بعدی

این پیام یک گزارش مربوط به DV است. در سطرهای بعدی شبکه و فاصله مان را از شبکه

مشخص می کنیم . مثلًا فاصله این نود از شبکه ۱۴۴.۲.۹.۰ و شبکه ۱۴۴.۲.۷.۰ بدون فاصله

و با هزینه ۰ است و فاصله تا شبکه ۱۴۴.۲.۱۲.۰ برابر ۱ است.

این جدول مربوط به نود ۱ R است چون به دو شبکه بدون واسطه وصل است و به شبکه

۱۴۴.۲.۱۲.۰ با یک گام و از طریق نود ۳ R متصل شده است.

## الگوریتم های مسیریابی در اینترنت

• بردار فاصله **Distance Vector**

• پروتکل **RIP**

• تایمراهای **RIP**

• دوره ای (**periodic**) : ۳۵-۲۵ ثانیه

• انقضا (**expiration**) : ۱۸۰ ثانیه

• جمع آوری زباله (**garbage collection**) : ۱۲۰ ثانیه

در شبکه چیز ثابتی وجود ندارد و توپولوژی ، بار ، لینکها و نودها در حال تغییر هستند، لذا پروتکلهای مسیریابی باید این تغییرات را رصد کنند و برای اینکار Rip از یک سری تایمر استفاده می کند و پس از انقضا تایمر ها ، فرایند مسیریابی را تکرار می کنند. برای این کار چند نوع تایمر وجود دارد.

تایمراهای دوره ای: در هر ۲۵-۳۵ ثانیه تکرار می شوند

تایمر انقضا: که هر ۱۸۰ ثانیه تکرار میشود

تایمر جمع آوری زباله: که هر ۱۲۰ ثانیه یکبار، اطلاعاتی که کاربرد ندارند را حذف می نماید.

## الگوریتم های مسیریابی در اینترنت

### • مشکلات RIP

- عدم در نظر گرفتن ظرفیت
- زمانبر بودن محاسبه تاخیرها
- کندی همگرایی
- ایجاد ترافیک به خاطر بسته های echo
- عدم بهینگی تاخیرها به خاطر عدم همزمانی
- عدم پایداری

Rip مشکلات

یک : عدم در نظر گرفتن ظرفیت : در این مسیریابی ، ظرفیت لینکها در نظر گرفته نمی شود و تاخیر ها لحاظ میشود

دو: زمانبر بودن محاسبه تاخیرها : که پروسه ای زمانبر است و اطلاعات کند به روز می شود

سه: کندی همگرایی

چهار : ارسال بسته های Echo که خود باعث ایجاد ترافیک می شود

پنج : به علت عدم همزمانی نودها ، محاسبه تاخیرها کاملا درست و بهینه نیست

شش: عدم پایداری یکی از مشکلات Rip است که در صفحه بعد توضیح داده شده است.

## الگوریتم های مسیریابی در اینترنت

### RIP مشکلات

• عدم در نظر گرفتن

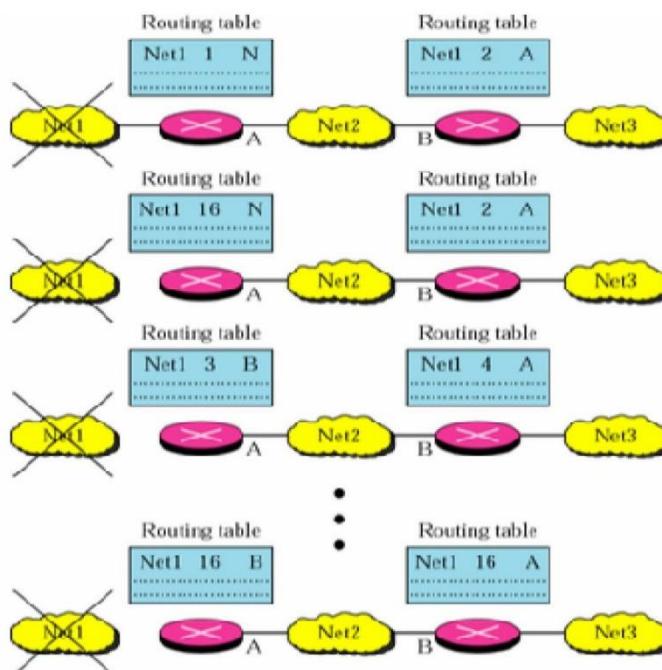
• زمانبودن محاسبه

• گندی همگرایی

• ایجاد ترافیک به خارج

• عدم بینگی تاخیرها

• عدم پایداری



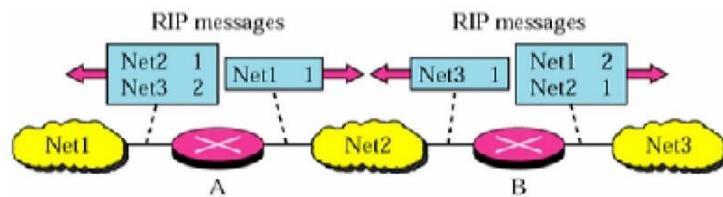
21 of 31

در مورد عدم پایداری ، زمانی که  $A$  از نود  $A$  قطع می شود ، روتیری که دورتر است در چه زمانی متوجه این قضیه می شود ؟ که در ۹ اسلاید قبل توضیح داده شد.(Count To Infinity)

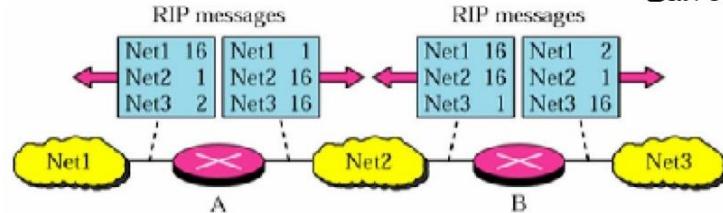
## الگوریتم های مسیریابی در اینترنت

### • راه حلهای عدم پایداری :

عدم ارسال اطلاعات از مسیریابهایی که از آنها دریافت شده



ارسال حداقل فاصله به مسیریابی که از طریق آن مسیر پیدا شده است



22 of 31

یکی از راه حل های جلوگیری از عدم پایداری استفاده از روش Split Horizon است که در اینجا نود ها مسیر را به سمت یک نود مشخص را به همسایه هایشان اعلام نمی کنند و فقط فاصله آنرا اعلام می کنند (که اگر مسیر را اعلام می کردند، در بعضی مواقع بهتر بود). در گفته میشود اطلاعات مسیریابی که از مسیرهای همسایه دریافت کرده را مجددا برای خود آن همسایه ارسال نکنید. مثلا زمانی که روتر B قصد دارد اطلاعات مسیریابی خود را به روتر A اعلام کند، نباید و لازم نیست که اطلاعات روتر A یعنی Net1 را به سمت روتر A ارسال کند و فقط اطلاعات Net3 را ارسال می کند.

راه حل دوم Poison Reverse است که مقدار بینهایت را برای مسیریابی می فرستد که شبکه Net<sup>۳</sup> را از طریق آن روتر پیدا کرده است. (در اینجا ۱۶ بینهایت است) مثلاً روتر B برای شبکه Net<sup>۳</sup> مقدار بینهایت را برای فاصله ارسال می کند. همچنین برای روتر A مقدار شبکه های Net<sup>۱</sup> و Net<sup>۲</sup> را بینهایت می فرستد چون اطلاعات را از خود روتر A گرفته است.

## الگوریتم های مسیریابی در اینترنت

### • نسخه ۲ RIP

#### • پشتیبانی از CIDR

Command	Version	Reserved
Family		Route tag
	Network address	
	Subnet mask	
	Next-hop address	
	Distance	

#### • کاربرد دیگر : authentication

Command	Version	Reserved
FFFF		Authentication type
	Authentication data 16 bytes	
	⋮	

23 of 31

در Rip Verstion<sup>۲</sup> ، از Cidr (مسیریابی دامنه های بدون کلاس) پشتیبانی می کند ، یعنی در این نسخه وجود دارد. در این پیام گفته می شود Next Hop Address و Subnetmask

برای رسیدن به Next Hop فلان از طریق Subnet Mask با Network Address باید اقدام شود، که در این حالت Distance بهمان است. در اینجا همسایه متوجه می شود که اگر Next Hop خود گیرنده پیام است، آنرا در نظر نگرفته و در محاسبات از آن استفاده نمی کند.

مورد دوم در Rip V2 بحث Authentication است که با احراز هویت پیامهایی که از سمت محاجم دریافت می شود را در نظر نگیرد ، تا موجب اختلال در شبکه و پروتکل نشود.

## الگوریتم های مسیریابی در اینترنت

### • حالت پیوند

#### ◦ یادگیری همسایه

#### ◦ اندازه گیری هزینه هر خط

#### ◦ ساخت بسته های link state

#### ◦ توزیع بسته های link state

#### ◦ محاسبه تاخیر

### • ظرفیت را نیز در نظر می گیرد

مسیر یابی Link State ویژگیها و مراحل زیر را دارد:

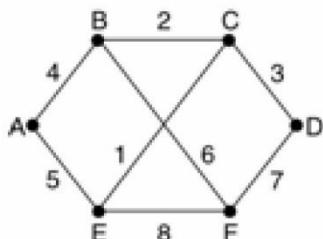
۱- در بحث یادگیری همسایه در پیامهایی که دریافت می کنیم اطلاعات کل شبکه وجود دارد .

۲- در Link State باشد هزینه Cost هر لینک محاسبه شود. ۳- سپس بسته های Link State Message ساخته می شود. ۴- و این بسته ها به تمام نودهای داخل شبکه ارسال می گردد. ۵- در انتهای محاسبه تاخیر صورت میگیرد. ۶- در Link State ظرفیت هر لینک ، به عنوان معیار در نظر گرفته می شود.

این موارد شش گانه تفاوت با Distance Vector نیز هست.

## الگوریتم های مسیریابی در اینترنت

### • بسته های link state



Link	State	Packets
A	Seq.	E
B	Seq.	F
C	Seq.	A
D	Seq.	B
E	Age	C
F	Age	D

در اینجا بسته های Link State ایجاد شده توسط هر نود نشان داده شده است . این بسته ها به سایر نودهای شبکه اعلام می گردد ( Advertise Seq No. و Age ). در این پیامها وجود دارد و پس از آن همسایه ها و فاصله آن تا همسایه نشان داده شده است. مثلا در نود A که دو

همسایه E,B دارد فاصله آن ۴ و ۵ است ( عدد ذکر شده ، هر معیاری از جمله ازدحام ، ظرفیت

لینک ، تاخیر و یا سایر پارامترهای هزینه می تواند باشد )

## الگوریتم های مسیریابی در اینترنت

### • توزیع بسته های link state

• سیل آسا (flooding)

• شماره ترتیب (sequence number)

• عمر بسته (age)

• تصدیق

توزیع بسته های Link State با روش های زیر انجام می گیرد: در اکثر پروتکل ها این توزیع به روش سیل آسا صورت می گیرد. یعنی هر نود که بسته ای را دریافت کرد آنرا به همه همسایه های خود ارسال می کند. در اینجا برای اینکه یک بسته دائما در داخل شبکه نچرخد از Seq No. استفاده می شود ، به این صورت که اگر نودی یک Seq No. را برای بار دوم دریافت کرد ، دیگر آنرا ارسال مجدد نمی کند. با این کار پس از مدتی بسته ها از داخل شبکه جمع می شود.

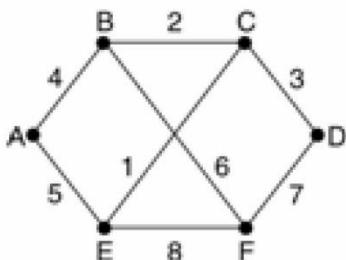
با فیلد Age از ارسال بسته ها به فوائل طولانی جلوگیری می کنیم . زیرا اگر قطر شبکه خیلی زیاد باشد ، نیاز به دانستن اطلاعات در فوائل خیلی دور نیست. در فیلد Age پس از مدتی بسته در شبکه Discard شده و دیگر Forward نمی شود . زمانی که یک نود بسته ای را با Seq

No. بخصوصی دریافت کرد باید Ack آنرا به فرستنده ارسال کند و در واقع آنرا تصدیق نماید

تا از پخش شدن بسته های خراب در شبکه جلوگیری شود.

## الگوریتم های مسیریابی در اینترنت

### ۰ توزیع بسته های link state



- سیل آسا (flooding)

- شماره ترتیب (sequence number)

- عمر بسته (age)

- تصدیق

Source	Seq.	Age	Send flags			ACK flags			Data
			A	C	F	A	C	F	
A	21	60	0	1	1	1	0	0	
F	21	60	1	1	0	0	0	1	
E	21	59	0	1	0	1	0	1	
C	20	60	1	0	1	0	1	0	
D	21	59	1	0	0	0	1	1	

Link State Propagation

27 of 31

پروسه فرستادن تصدیق با تاخیر انجام می شود. و فقط به آن سمتی تصدیق ارسال می شود که از آن سمت بسته دریافت شده باشد . در این جدول نحوی انتشار بسته Link State آمده است. در جدول برای Source های مختلف در شبکه یک Ack Flag و یک Send Flag به سمت همسایه ها وجود دارد. جدول فوق مربوط به نود B است ، که در سطر اول بسته های را از نود A دریافت کرده و باید آنرا به سایر نودهای همسایه ارسال کند . لذا برای نود A یک  $Ack=1$  می

فرستد و برای یک  $F, C$  می فرستد. در سطر سوم نود  $B$  بسته ای را از  $E$  دریافت کرده و چون مسیر غیر مستقیم است و از طریق  $A, F$  دریافت کرده پس  $Ack$  را برای این دو نود می فرستد و برای نود  $C$  هم  $Send$  را می فرستد.

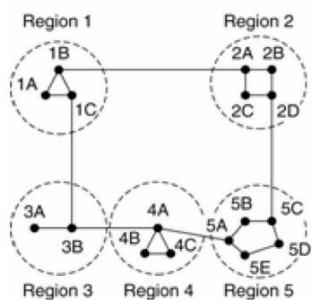
لذا زمانی که یک نود بسته ای را دریافت کرد ، مدتی صبر می کند و اگر از مسیر دیگری همان پیام را دریافت نمود ، برای آن نودها  $Ack$  می فرستد و دیگر بسته ای را برای آنها ارسال نمی کند و فقط برای همسایگانی که از آنها این بسته را نگرفته اند ، آنرا می فرستد . اگر هر سه  $Send$  صفر شود یعنی از همه همسایه ها این بسته را گرفته و نیاز به ارسال مجدد آن نیست.

## الگوریتم های مسیریابی در اینترنت

### • توزیع بسته های link state

#### • اشکال : بزرگی جداول

#### • راه حل : ایجاد سلسله مراتب



(a)

Full table for 1A		
Dest.	Line	Hops
1A	-	-
1B	1B	1
1C	1C	1
2A	1B	2
2B	1B	3
2C	1B	3
2D	1B	4
3A	1C	3
3B	1C	2
4A	1C	3
4B	1C	4
4C	1C	4
5A	1C	4
5B	1C	5
5C	1B	5
5D	1C	6
5E	1C	5

(b)

Hierarchical table for 1A		
Dest.	Line	Hops
1A	-	-
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

(c)

یکی از اشکالات Link State در این است که چون تعداد مسیریابها در شبکه زیاد است لذا سطرهای جدول مسیریابی زیاد خواهد شد و بزرگی جدول یکی از اشکالات Link State است. راه حل این مشکل ایجاد سلسله مراتب است. یعنی روترهای نزدیک به هم را در قالب یک ناحیه (Region) طبقه بندی شوند.

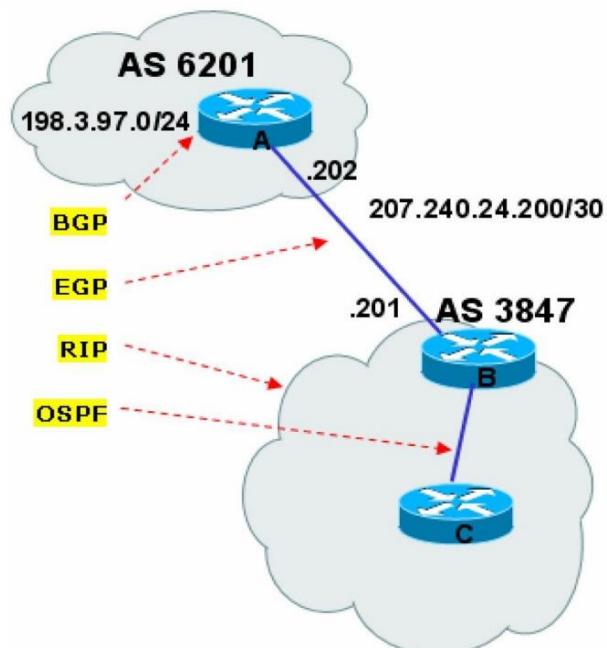
جدول (B) مربوط به روتر ۱a است و جدول (C) پس از انجام ناحیه بندی است، که تعداد سطرهای خیلی کم شده است. در اینجا فقط مسیر یابی به روترهای داخل ناحیه را داریم و برای سایر مسیرها فقط Region مقصد را مشخص می کنیم. یعنی برای Dest:Region<sup>۵</sup> از طریق Cost:۱c و با Nexthop: ۱c است.

۱a — ۱c — Reg<sup>۳</sup> — Reg<sup>۴</sup> — Reg<sup>۵</sup> (۴ Step To Dest )

در این حالت رسیدن به سایر نودهای Region<sup>۵</sup> از طریق همین روش انجام می پذیرد. برای اینکه تعداد گامها و Next Hop C را متوجه شویم به جدول (B) رفته و مسیر را از روی آن تشخیص می دهیم. یعنی از ۱a به ۵a از نod ۱c عبور می کند و از ۱a به ۵c از مسیر ۱b می رود. در جدول (C) ما به شبکه به صورت منطقه ای نگاه می کنیم و به داخل آن کاری نداریم، یعنی از ۱a به کل Region<sup>۵</sup> از طریق ۱c و Hop=۴ حرکت می کنیم. لذا در اینجا تعداد گامها تعداد Region های بین راه است که در نتیجه گامهای کمتری است.

## الگوریتم های مسیریابی در اینترنت

### • سلسله مراتب link state



29 of 31

اینترنت مجموعه ای از AS ها است که به یکدیگر متصل است. در اینجا ما مسیر یابی درون As و بین As را داریم. مسیر یابی درون As ، ospf و rip است و در لبه As ها از مسیر یابی Bgp و در خارج و بین As ها ، مسیر یابی Egp را داریم .

# الگوریتم های مسیریابی در اینترنت

## OSPF •

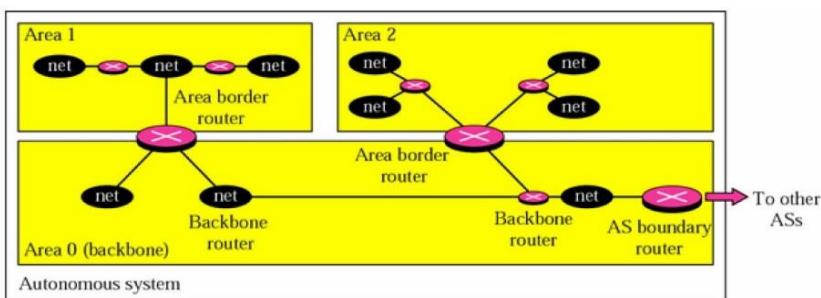
### • چهار دسته مسیریاب در داخل AS

#### • مسیریاب های داخل نواحی

#### • مسیریاب های لبه های نواحی

#### • مسیریاب های ستون فقرات

#### • مسیریاب های مرز AS



30 of 31

در داخل AS ها ماقچهار دسته مسیریاب داریم. در شکل بالا AS سلسله مراتبی (ناحیه بندی شده) داریم که به سه Region یا Area تقسیم شده است. دسته اول مسیریابها در داخل نواحی قراردارند و به سایر نواحی دسترسی ندارند (صورتی کوچک). دسته دوم مسیریابها در لبه های نواحی قرار دارند که به آن Area Boarder Router گفته می شود. دسته سوم AS هستند و دسته چهارم مسیریابی مرز Backbone Router می گویند، که اتصال AS ها را به یکدیگر برقرار می کنند.

## جلسه هفتم:

### الگوریتم های مسیریابی در اینترنت

#### OSPF •

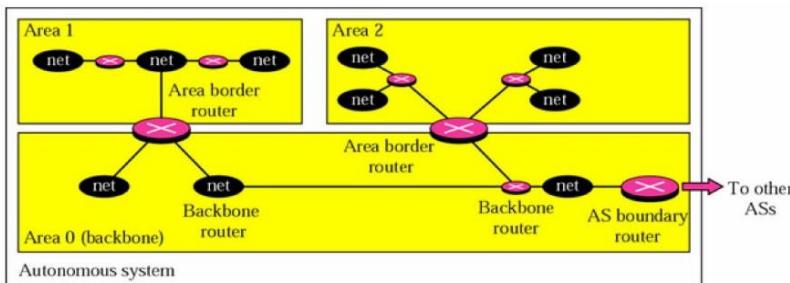
##### • چهار دسته مسیریاب در داخل AS

- مسیریاب های داخل نواحی

- مسیریاب های لبه های نواحی

- مسیریاب های ستون فقرات

- مسیریاب های مرز AS



4 of 31

پروتکل های مسیریابی در شبکه (مرور مجدد این اسلاید)

: OSPF

چهار دسته مسیریاب در داخل AS در OSPF داریم:

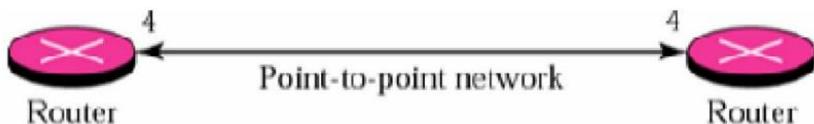
- مسیریابهای داخل نواحی : از آنجایی که یک AS به نواحی مختلفی تقسیم می شود یکسری از مسیریابها فقط داخل یک ناحیه هستند و ارتباطی با بیرون ندارند.
- مسیریابهای لبه های نواحی : یکسری از مسیریابها در لبه نواحی هستند که کسیریابها را به هم وصل میکنند.

- مسیریابهای ستون فقرات: یکسری میسریابهای Back Bone هستند که ارتباطات اصلی را داخل AS برقار میکنند و ترافیک بیشتری روی آنها است.
- مسیریابهای مرز AS : ارتباط یک AS با دیگر AS را برقار میکنند.

## الگوریتم های مسیریابی در اینترنت

### ۰. انواع اتصالات در AS

#### Point to point •



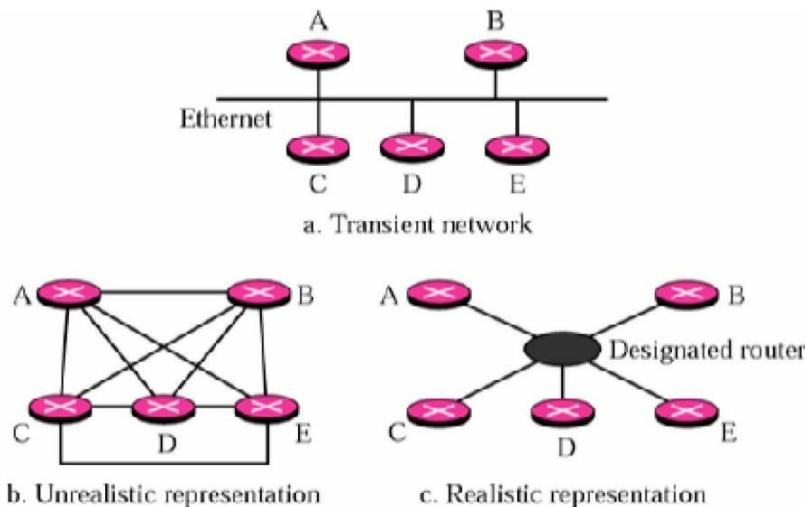
: انواع اتصالات در AS

- که یک روترا به یک روترا دیگر به صورت مستقیم وصل میکند.

# الگوریتم های مسیریابی در اینترنت

## • انواع اتصالات در AS

### • ارائه اطلاعات شبکه توسط یک مسیریاب خاص : Transnet •



6 of 31

• Transient : اطلاعات شبکه توسط یک مسیریاب خاص ارائه میشود. مثلا در اترنت شکل

A اطلاعات شبکه بین روترا میتواند برقرار شود. درواقع در اینجا ما ارتباط بین همه روترا را داریم یعنی اگر بخواهیم مدل این ارتباطات را به صورت ارتباطات Point To Point نمایش دهیم مدل شکل B خواهد شد یعنی همه روترا با هم ارتباط دارند مثلا با همه روترا ارتباط دارد و B هم همینطور. در واقع یک Mesh کامل است. در واقع شکل B یک نمایش غیر واقعی (Unrealistic Representation) از ارتباطات درون شبکه است. در Transient Network (شکل A) است.

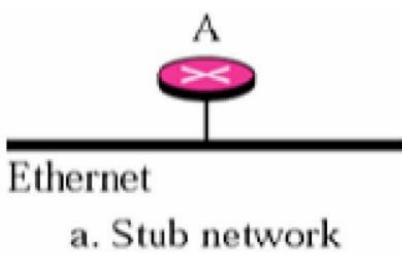
B) ارتباطات بین روترا یک Mesh کامل خواهد شد ولی به صورت لاجیکال است، به صورت فیزیکال در واقع ارتباطات به صورت شکل A است یعنی درواقع شبکه توپولوژی Bus دارد و روترا همه با هم ارتباط دارند. و نمایش یک مقدار واقعی تر اینگونه میتواند

باشد که ما یک روتر مجازی در وسط داشته باشیم و همه روترهای آن وصل شوند (شکل C) و ارتباطات همه روترهای آن روتر میتوانند برقرار شود.

## الگوریتم های مسیریابی در اینترنت

### • انواع اتصالات در AS

#### • فقط یک مسیریاب در شبکه وجود دارد



a. Stub network



b. Representation

#### • virtual و

- Stub : حالتی در شبکه که فقط یک روتر وجود دارد و بقیه نودهای شبکه Host ها هستند که به این روتر متصل میشوند و نمایش آن به صورت شکل بالا میباشد که مثلا یک Designated Router یا این روتر متصل است.
- Virtual : راجع به آن صحبت خواهیم کرد...

## الگوریتم های مسیریابی در اینترنت

### • انواع اعلان وضعیت اتصال (link state advertisement)

- **LSA Type 1: OSPF Router LSA**
- **LSA Type 2: OSPF Network LSA**
- **LSA Type 3: OSPF Summary LSA**
- **LSA Type 4: OSPF ASBR Summary LSA**
- **LSA Type 5: OSPF ASBR External LSA**
- **LSA Type 6: OSPF Group Membership LSA**
- **LSA Type 7: OSPF Not So Stubby Area (NSSA) External LSA**
- **LSA Type 8: OSPF External Attributes LSA (OSPFv2) / Link Local LSA (OSPFv3)**
- **LSA Type 9: OSPF Link Scope Opaque (OSPFv2) / Intra Area Prefix LSA (OSPFv3)**
- **LSA Type 10: OSPF Area Scope Opaque LSA**
- **LSA Type 11: OSPF AS (Autonomous System) Scope Opaque LSA**

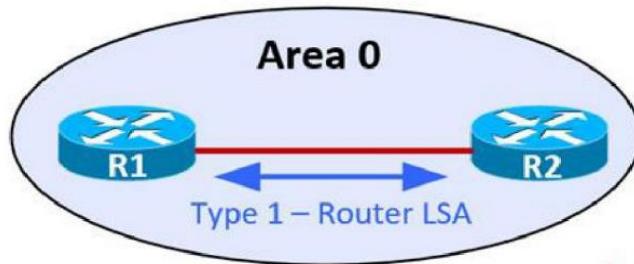
در OSPF روتراها باید وضعیت اتصال را به هم اعلام کنند تا مسیریابی امکان پذیر باشد. (قبل اگفتیم OSPF مبتنی بر Link State است و در این حالت هر روتر باید وضعیت اتصال را به همه روتراهای دیگر در شبکه اطلاع دهد تا هر روتر توبولوژی کل شبکه را داشته باشد تا بتواند براساس آن تصمیم گیری کند و الگوریتم Dijkstra را اجرا کند و مسیر به هر نod دیگر در شبکه را بدست آورد).

برای اعلان وضعیت، انواع بسته های اعلان وضعیت داریم (Link State Advertisement) یا LSA که ۱۱ نوع است و در شکل بالا فقط عنوان آنها آمده و در اسلایدهای بعدی به آنها میپردازیم.

# الگوریتم های مسیریابی در اینترنت

## • انواع اعلان وضعیت اتصال (link state advertisement)

### Router LSA •

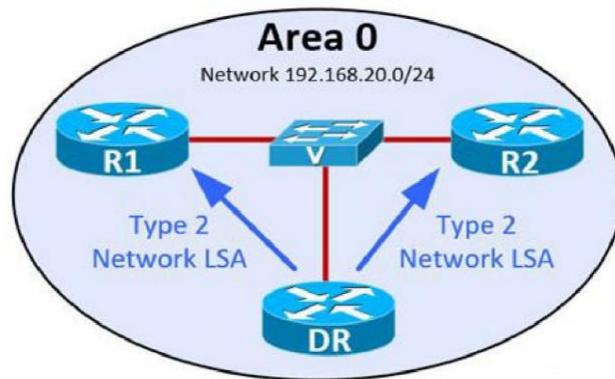


Router LSA : به اعلان وضعیت روترهای درون یک Area درواقع Router LSA میگوییم. گفته‌یم که AS را به Area های مختلفی تقسیم بندی میکنیم (مربوط به بحث Region بندی بود که گفته‌یم برای اینکه جدول مسیریابی بزرگ نشود و مسیریابی راحت‌تر انجام شود میتوانیم کل شبکه را به بخشها و نواحی مختلف تقسیم کنیم. که این Area ها به این منظور در OSPF در نظر گرفته میشوند) و روتربی که فقط داخل یک Area هست و ارتباطی با جای دیگر ندارد را روتر داخل Area میگفته‌یم که وقتی وضعیت لینک خودش را به روتربی داخل همان Area میدهد اینکار را براساس Type Router LSA انجام میدهد.

## الگوریتم های مسیریابی در اینترنت

### • انواع اعلان وضعیت اتصال (link state advertisement)

#### Network LSA •

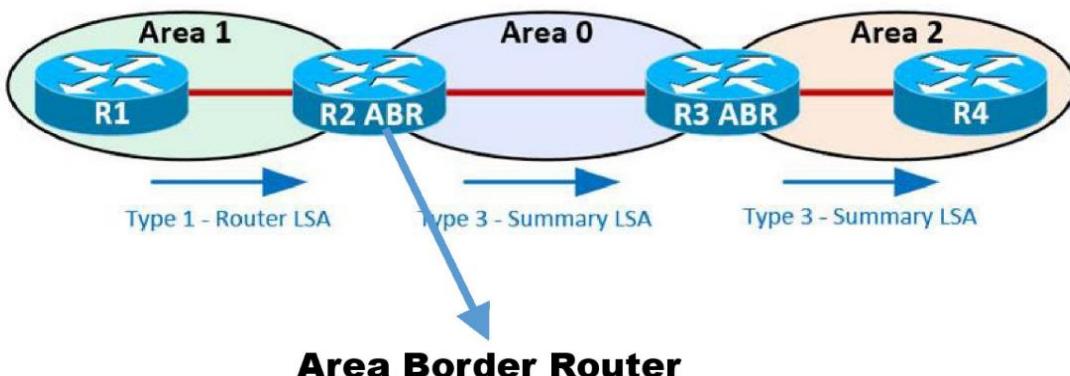


Network LSA یا روتر منتخب از طریق Designated Router : Network LSA •  
وضعیت را اعلام میکند.

## الگوریتم های مسیریابی در اینترنت

### • انواع اعلان وضعیت اتصال (link state advertisement)

#### SUMMARY LSA •

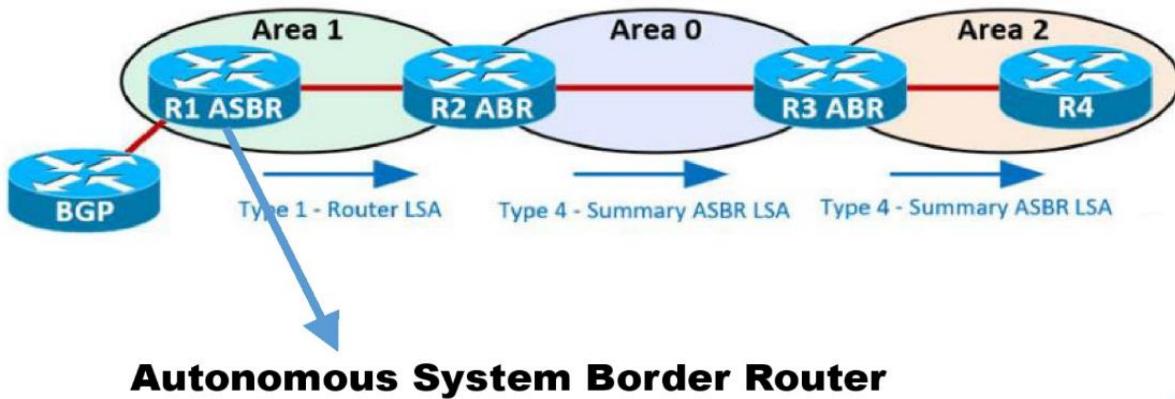


- وقتی روتری که در مرز Area هست (Area Border Router) Summary LSA میخواهد به روترهای داخل Area یا وضعیت خارج از Network را اعلام میکند از Summary LSA استفاده میکند.

## الگوریتم های مسیریابی در اینترنت

- انواع اعلان وضعیت اتصال (link state advertisement)

### ASBR SUMMARY LSA •

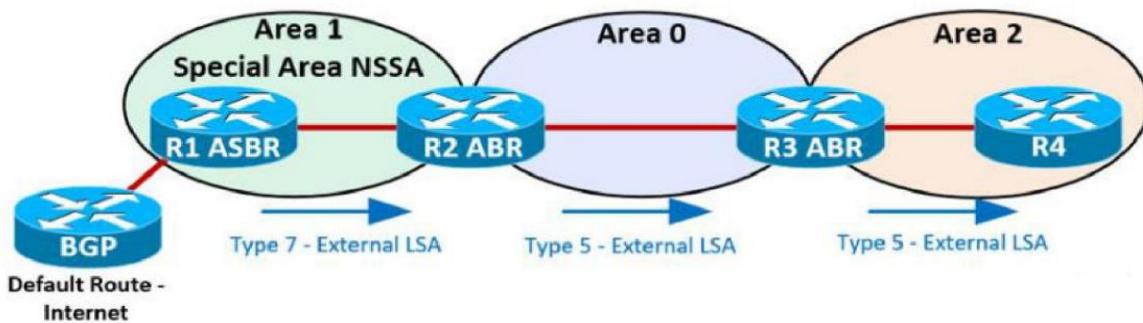


- وقتی روتری که در مرز AS (AS Border Router) ASBR Summary LSA است اعلان وضعیت میکند اینکار را از طریق ASBR Summary LSA انجام میدهد.

# الگوریتم های مسیریابی در اینترنت

## ۰. انواع اعلان وضعیت اتصال (link state advertisement)

### ASBR EXTERNAL LSA •

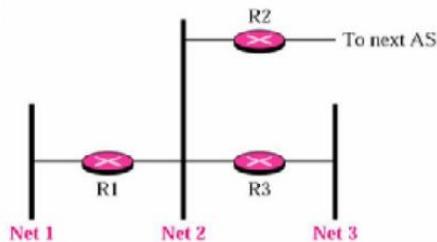


برای ارسال دیتای External به داخل شبکه است. یعنی ASBR External LSA •  
وقتی AS Border Router وضعیت خارج شبکه را اعلام بکند LSA میشود.

## الگوریتم های مسیریابی در اینترنت

### ۰ انواع اعلان وضعیت اتصال (link state advertisement)

مثال: در تصویر زیر معین کنید هر مسیریاب چه Router Link LSA هایی را ارسال می کند.

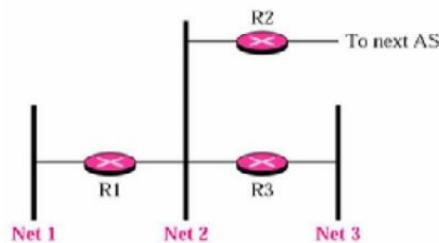


میخواهیم ببینیم در شکل بالا هر مسیریاب چه Router Link LSA ای را ارسال میکند. آیا میتوانید متوجه شوید هر کدام از این روتراها به چه نت هایی وصل هستند؟

## الگوریتم های مسیریابی در اینترنت

### ۰ انواع اعلان وضعیت اتصال (link state advertisement)

مثال: در تصویر زیر معین کنید هر مسیریاب چه Router Link LSA هایی را ارسال می کند.



همه مسیریابها Router Link LSA ها را اعلان می کنند.

R1 دو اتصال دارد: Net1 و Net2

R2 یک اتصال دارد: Net2

R3 دو اتصال دارد: Net2 و Net3

همه مسیریابها Router Link LSA ها را اعلام میکنند یعنی در این روترا رو تری نیست که نیاز نباشد Link LSA را اعلان کند.

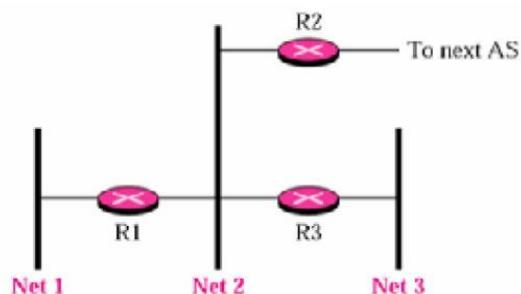
R<sup>۱</sup> دو تا اتصال دارد یعنی وضعیت دو تا لینک خودش به Net<sup>۱</sup> و Net<sup>۲</sup> را اعلان میکند.

R<sup>۲</sup> یک اتصال به Net<sup>۲</sup> دارد و آنرا اعلان میکند.

R<sup>۳</sup> دو تا لینک به Net<sup>۲</sup> و Net<sup>۳</sup> دارد و آن را اعلان میکند.

## الگوریتم های مسیریابی در اینترنت

### انواع اعلان وضعیت اتصال (link state advertisement)

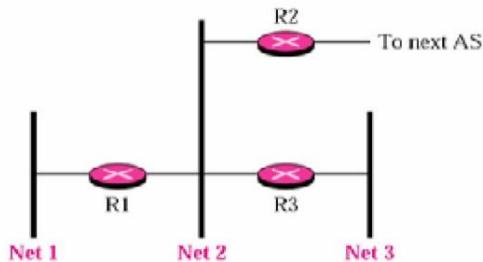


کدام مسیریاب اطلاعات Network Link LAS را به خارج ارسال می کند؟

جواب سوال بالا مشخص است روتری که به خارج از شبکه دسترسی دارد.

# الگوریتم های مسیریابی در اینترنت

## • انواع اعلان وضعیت اتصال (link state advertisement)



کدام مسیریاب اطلاعات Network Link LAS را به خارج ارسال می کند؟

همه سه شبکه باید Network Link ها را اعلان نمایند.

اعلان Net1 بوسیله R1 انجام می گیرد، زیرا تنها مسیریاب متصل به آن است و بنابراین Designated Router می باشد.

اعلان Net2 بوسیله R1 یا R2 و یا R3، براساس اینکه کدامیک از آنها به عنوان Designated Router انتخاب شود، می تواند انجام گیرد.

اعلان Net3 بوسیله R3 انجام می گیرد، زیرا تنها مسیریاب متصل به آن است و بنابراین Designated Router می باشد.

17 of 31

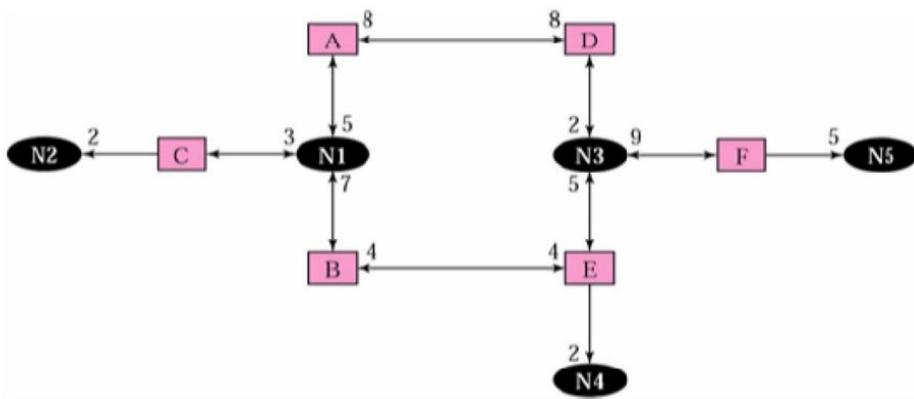
در مثال بالا اگر Designated Router برای Net<sup>۲</sup>، R<sup>۱</sup> باشد، دیگر R<sup>۲</sup> چیزی اعلان نمیکند.

# الگوریتم های مسیریابی در اینترنت

## OSPF •

• همه مسیریاب ها پایگاه حالت پیوند یکسان دارند

• مثال: نحوه محاسبه کوتاهترین مسیر در شبکه زیر

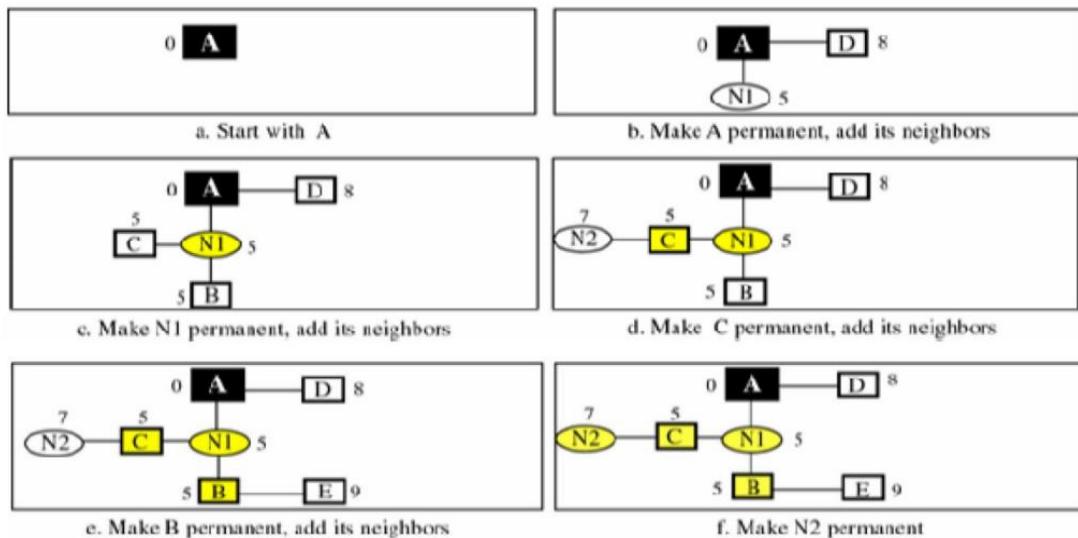


گفتیم OSPF مبتنی بر Link State است و همه مسیریاب ها پایگاه داده حالت پیوند یکسان دارند. وقتی وضعیت اتصالات را روتراها به هم اطلاع می دهند، با فرض اینکه تغییری در شبکه نداشته باشیم، بعد از مدتی همه مسیریابهای داخل شبکه اطلاعات مشابهی خواهند داشت و اگر تغییری اعمال شود این تغییرات باید در شبکه آپدیت شود و دوباره بعد از مدتی پایگاه داده حالت پیوند همه روتراها یکسان خواهد شد.

در مثال شکل بالا تعدادی روتر داریم از A تا F و تعدادی Net داریم از 1 تا 5.

# الگوریتم های مسیریابی در اینترنت

## OSPF •



در هر مرحله بین همسایگان کوتاهترین مسیر با کمترین وزن انتخاب میشود.

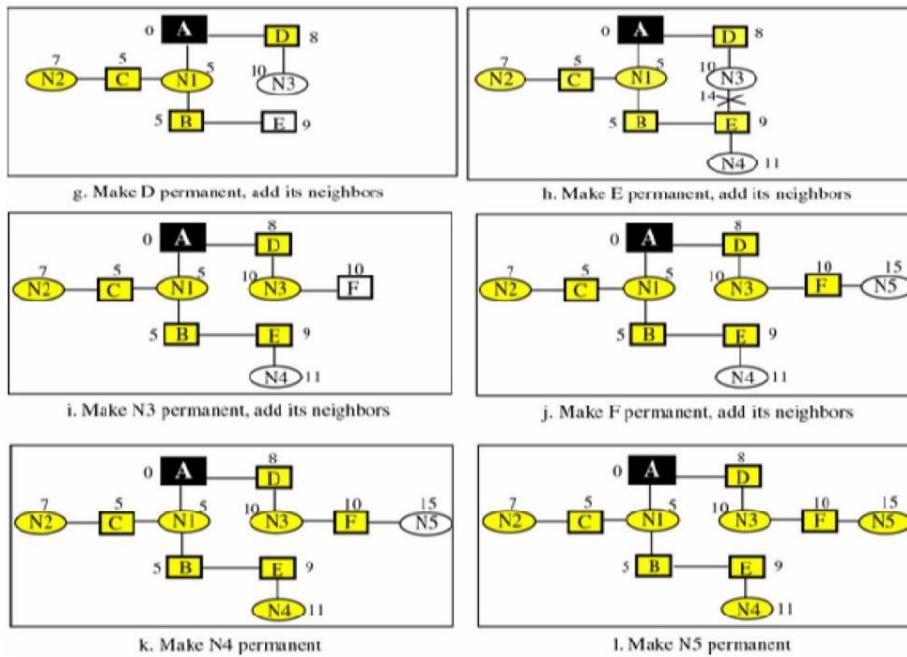
فرض کنیم میخواهیم برای A کوتاهترین مسیرها را حساب میکنیم پس از A شروع میکنیم. A را Permanent میکنیم و همسایگانش را Add میکنیم یعنی N<sup>1</sup> و D. اینجا بین N<sup>1</sup> و D (یعنی بین 5 و 8) از آنجا که N<sup>1</sup> کوچکتر است به عنوان Permanent انتخاب میشود و همسایگانش Add میشود یعنی C و B و حالا C انتخاب میشود چون کوچکتر است. همسایگان C اضافه میشود که N<sup>2</sup> هست.

در شکل D نودهایی که Permanent نشده اند B و N<sup>2</sup> و D هستند. و چون B کوچکتر است (استاد میگوید نمیدانم چرا عدها را نسبت به شکل اصلی تغییر داده اینجا!!!) B را Permanent کرده و همسایه هایش را اضافه کرده.

در شکل E بین N<sup>2</sup> و D و E و D که کوچکتر است را Permanent کرده.

# الگوریتم های مسیریابی در اینترنت

## OSPF •



20 of 31

و در شکل G بین D و E، کوچکترین یعنی D را Permanent کرده و همسایه آن یعنی N<sup>3</sup> را اضافه کرده است.

در شکل H بین N<sup>3</sup> و E، E را Permanent کرده و همسایه آن یعنی N<sup>4</sup> را اضافه کرده است. (N<sup>3</sup> چون قبلاً اضافه شده است دیگر آنرا اضافه نمیکنیم).

در شکل I، N<sup>3</sup> را Permanent کرده و همسایه آن یعنی F را اضافه کرده است.

در شکل J را FJ را Permanent کرده و همسایه آن N<sup>5</sup> اضافه میشود.

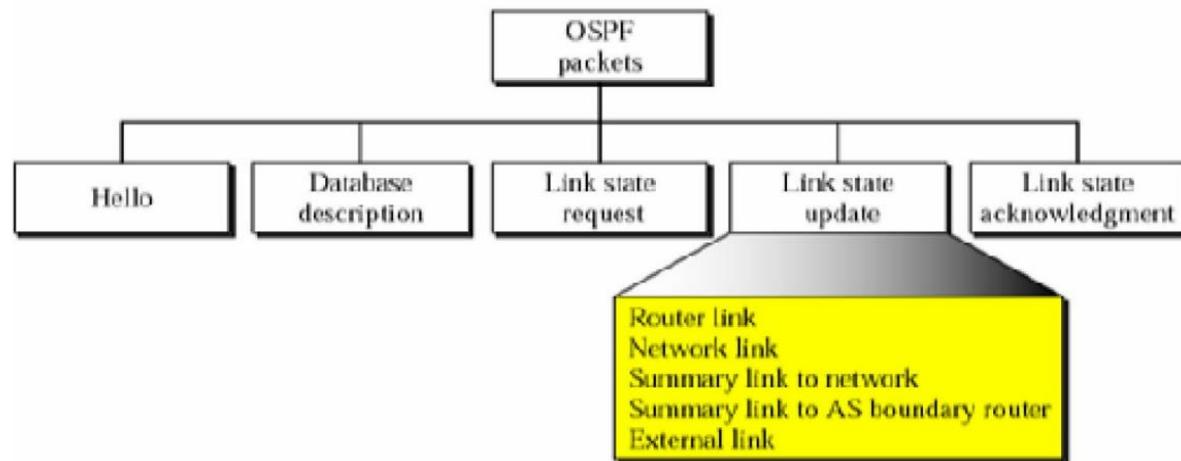
در شکل K و L، به ترتیب N<sup>4</sup> و N<sup>5</sup> به عنوان Permanent انتخاب میشوند ولی چون همسایه ای ندارند نمیتوانند اضافه نمیشود.

نودهایی که هنوز Permanent نشده اند ولی Add شده اند سفید رنگ نمایش داده میشوند و آنهایی که Permanent شده اند زرد رنگ نمایش داده میشوند. این کار تا زمانیکه تمام نودهای شبکه زرد رنگ شوند ادامه پیدا میکند.

در آخر اگر از نود اولی که شروع کردیم ارتباط را به دیگر نودها نگاه کنیم یک درخت بوجود می آید و این درخت در واقع کوتاهترین مسیر به تمام نودهای شبکه است. تقریباً این روش شبیه روش دایجسترا است.

## الگوریتم های مسیریابی در اینترنت

### • انواع بسته های OSPF



در این اسلاید انواع بسته های OSPF نمایش داده شده اند.

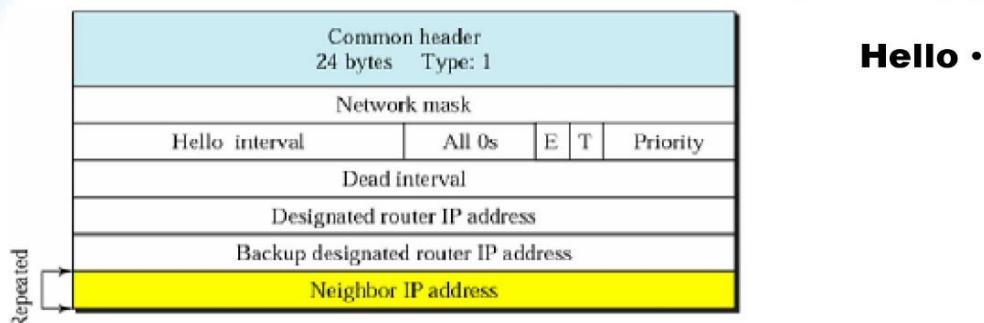
یک Hello Packet OSPF دارد برای اینکه نودها را انجام دهند و Neighbor Discovery همسایه هایشان را بشناسند.

بسته های OSPF شامل بسته های Link State Request و Database Description و Link State Update (که راجع به آن الان کمی صحبت کردیم) و

(وقتی بسته های Link State را نودها دریافت میکنند باید Link State Acknowledgment بدهند که این کار را با Link State Acknowledgment انجام میدهند) Acknowledge میباشد.

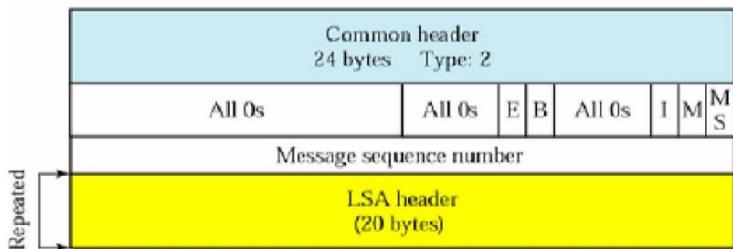
## الگوریتم های مسیریابی در اینترنت

### • انواع بسته های OSPF



**Hello** •

### • Database description

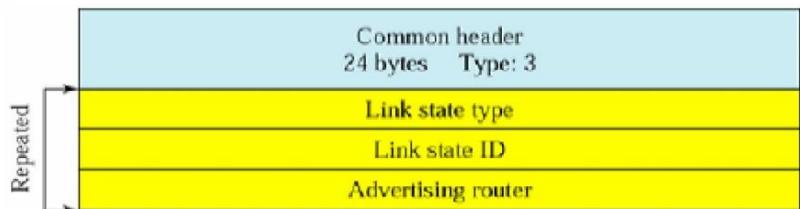


22 of 31

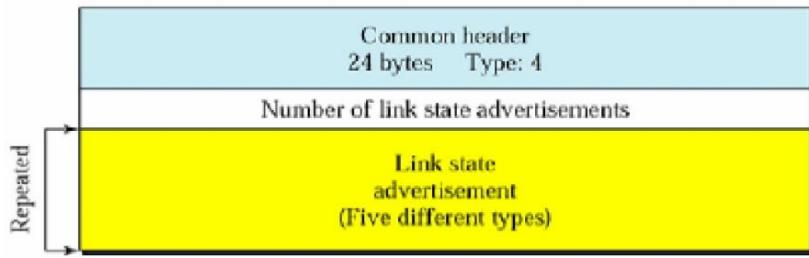
# الگوریتم های مسیریابی در اینترنت

## • انواع بسته های OSPF •

### Link state request •



### Link state update •



23 of 31

در این اسلایدها قالب بسته های مختلف را میبینید که ما راجع به این قالبها صحبتی نمیکنیم ولی شما میتوانید آنها را مطالعه کنید.

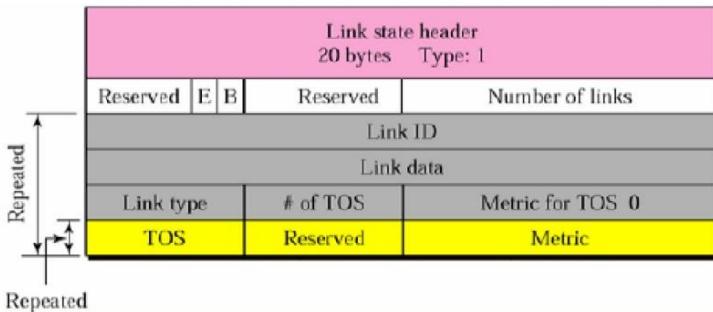
# الگوریتم های مسیریابی در اینترنت

## • انواع بسته های OSPF •

### LSA header format •

Link state age	Reserved	E	T	Link state type
Link state ID				
Advertising router				
Link state sequence number				
Link state checksum	Length			

### Router LSA •



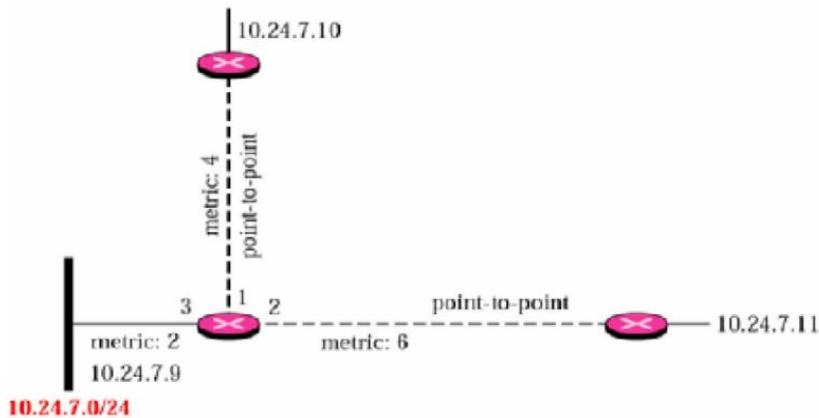
24 of 31

در این اسلاید نیز Router LSA و Header Format را میبینید. (جزئیات این بسته ها را من توضیح نمیدهم ولی خودتان بخوانید احتمال اینکه در امتحان بباید وجود داردا)

# الگوریتم های مسیریابی در اینترنت

## ۰. انواع بسته های OSPF

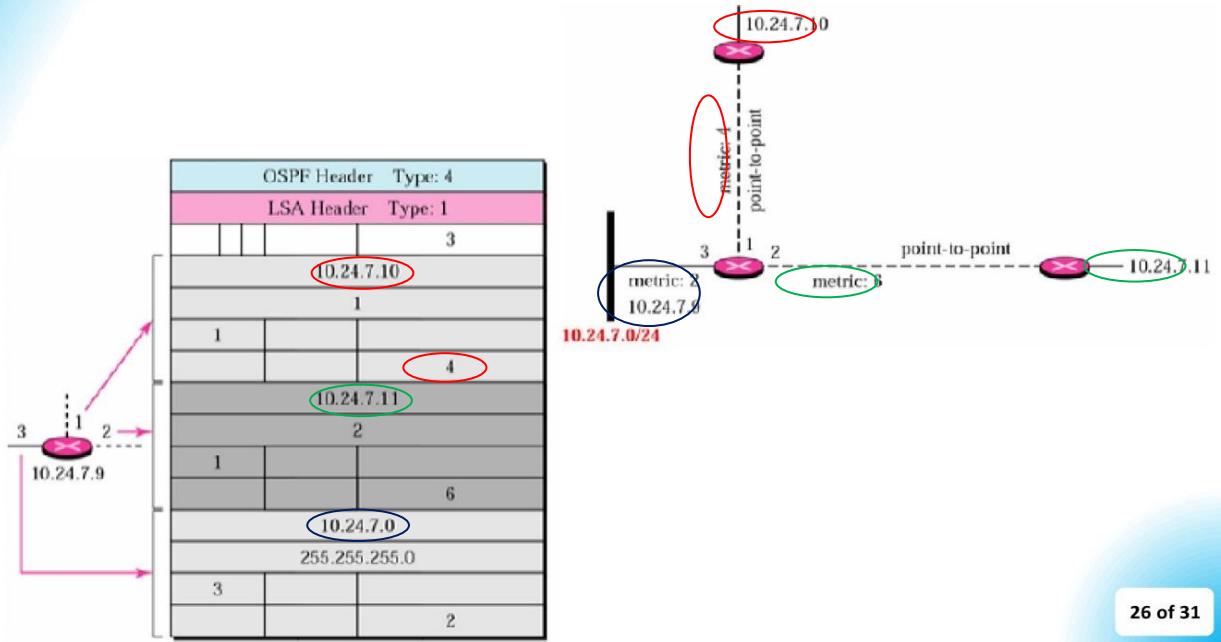
مثال: Router Link LSA ارائه شده توسط مسیریاب 10.24.7.9 چیست؟



این سوال میخواهد دسته Link State Advertisement ای که روتر 10.24.7.9 میفرستد را مشخص کند. در تصویر مشخص است که به سه تا نود یا شبکه وصل است. یکی به 10.24.7.10 و یکی به 10.24.7.11 وصل است و پورت دیگر به 10.24.7.0 متصل است بنابراین ارتباطات این ۳ تا لینک را باید در بسته بگذارد.

# الگوریتم های مسیریابی در اینترنت

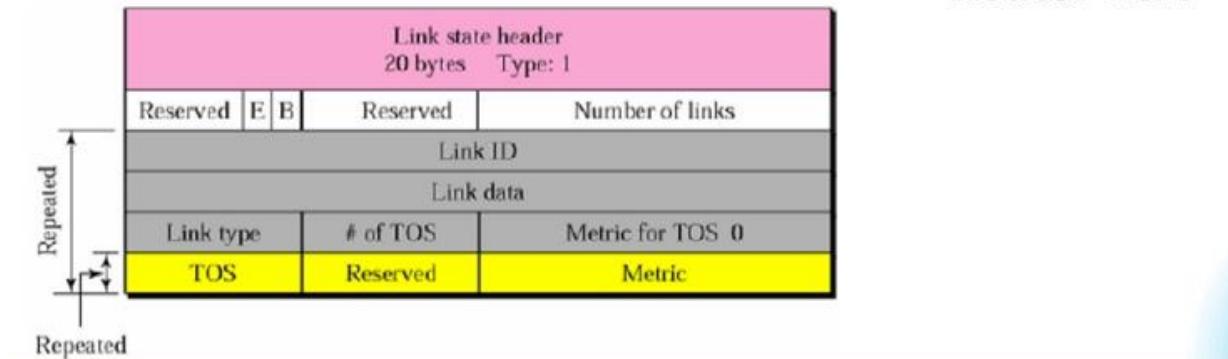
## ۰ انواع بسته های OSPF



بسته LSA ای که تولید میکند بعد از IP شبکه مورد نظر را گذاشته (استاد در جدول بالا به خط ۱۰.۲۴.۷.۱۰ اشاره میکند) و فیلدهای دیگر را پر کرده و متريک آن لينک را هم گذاشته (اشاره به عدد ۴) که مثلا ۱۰.۲۴.۷.۱۰ وزنش يا Cost اش يا متريکش ۴ است.

يا مثلا در ۱۰.۲۴.۷.۱۱ شماره يا Type ۲ است و متريکش ۶ است.  
و در ۱۰.۲۴.۷.۰ که يك شبکه است بنابراین Subnet Mask میخواهد و متريکش ۲ است.

## Router LSA •

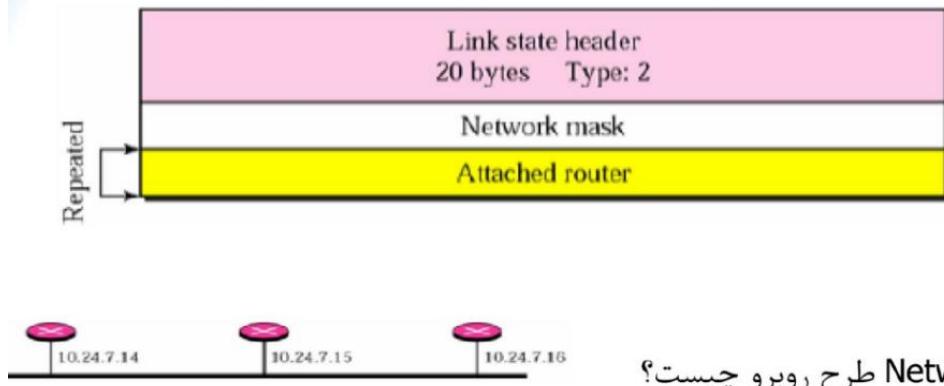


در چند اسلاید قبل جایی که Router LSA را گفته میبینیم که Link Data و Link ID را گذاشته میباشد (که برای روتراها شماره پورت و برای شبکه Subnet Mask است) گذاشته و همچنین Link Type را نیز گذاشته است.

## الگوریتم های مسیریابی در اینترنت

### • انواع بسته های OSPF

#### Network LSA •



مثال: Network Link LSA طرح رویرو چیست؟

قالب بسته Network LSA را در شکل بالا میبینید.

برای مثال بالا باید یک Network Mask و ۳ تا روتراهایی که Attach شده به این شبکه را بنویسید.

چون Network Mask در نظر بگیریم Class Less داریم باید Subnet Mask از این IP آدرسها به این ۲۵۵.۲۵۵.۲۵۵.۰ میتواند باشد ولی از Subnet Mask نمیرسیم. (در کتاب همانطور که در اسلاید بعدی میبینید ۲۵۵.۲۵۵.۲۵۵.۰ در نظر گرفته است که البته اشتباه نیست مگر اینکه Class Full باشد یا میدانیم ۸ بیت سمت راست را برای Host گذاشته ایم و همه Id ها را اینجا نیاورده ایم.)

اگر Class Less در نظر بگیریم باید Long Sperfix در نظر بگیریم و در اینجا ۱۰.۲۴.۷ مشترک است و اعداد قسمت کم ارزش ۱۴، ۱۵، ۱۶ است که بزرگترین ۱۶ است و میشود ۵ بیت (۱۶ میشود ۱۰۰۰۰ که میشود ۵ بیت). پس ۵ بیت سمت راست باید صفر باشد و ۳ بیت باید یک باشد و میشود ۱۱۱۰۰۰۰ و در هگز میشود E۰ و در دسیمال میشود ۲۲۴ پس ۱۰.۲۴.۷.۱۴ میشود Subnet Mask و روترها هم که در تصویر است :

۱۰.۲۴.۷.۱۵ ، ۱۰.۲۴.۷.۱۶

## الگوریتم های مسیریابی در اینترنت

### • انواع بسته های OSPF



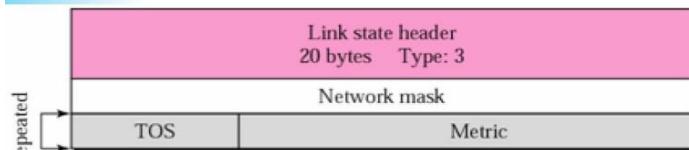
مثال: Network Link LSA طرح روبرو چیست؟

OSPF Header	Type: 4
LSA Header	Type: 2
255.255.255.0	
10.24.7.14	
10.24.7.15	
10.24.7.16	

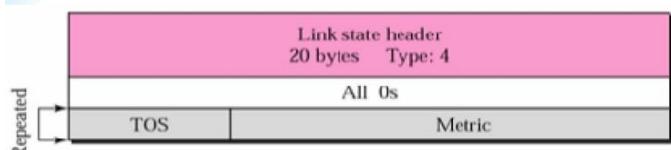
پس Network Link LSA برای این شبکه به این شکل خواهد بود.

## الگوریتم های مسیریابی در اینترنت

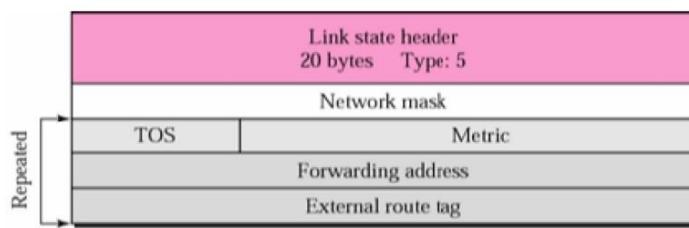
### • انواع بسته های OSPF •



### • ASBR summary LSA •



### • ASBR external LSA •



29 of 31

انواع بسته های OSPF را در شکل بالا میبینید.

برای مثلا Summary LSA اگر Metric و Network Mask را بنویسید کافی است.

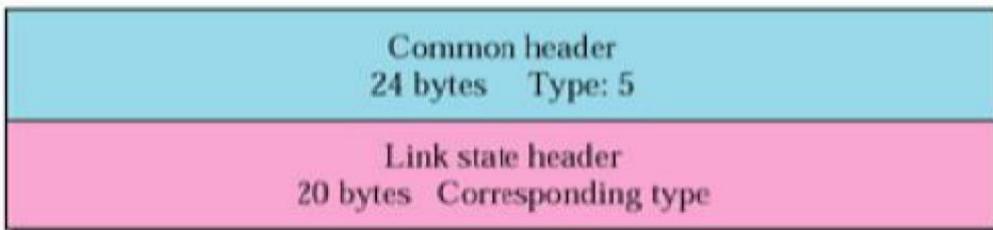
ASBR Summary LSA که فیلد آن همه صفر است.

در ASBR External LSA Forwarding و Metric و Network Mask اطلاعات وجود دارد. Address و External Route Tag

## الگوریتم های مسیریابی در اینترنت

### • انواع بسته های OSPF

#### Link state acknowledgment •



### • بسته های IP قرار می گیرند

در وجود دارد و بحث Link State Header هم Link State Acknowledgement در آن مطرح است.

بسته های IP قرار میگیرند که به نظر میرسد یک لایه بالاتر از IP است ولی در واقع اینگونه نیست. بسته های OSPF در واقع بسته های لایه شبکه هستند ولی در IP Datagram قرار میگیرند.

## پروتکل های مسیریابی

الگوریتم های مسیریابی در اینترنت  
مسیریابی برون AS  
مسیریابی چندپخشی (Multicast)  
مقدمه  
چندپخشی در یک شبکه فیزیکی

در این قسمت راجع به پروتکل های مسیریابی بروز AS صحبت میکنیم و بعد از آن به مسیریابی چند بخشی میپردازیم.

پروتکلی که در مسیریابی بروز AS مورد استفاده قرار میگیرد پروتکل BGP است.

## الگوریتم های مسیریابی در اینترنت

### • مسیریابی بین AS ها : BGP

#### • انواع AS

##### • Stub AS

◦ فقط یک راه ارتباطی به سایر AS ها

◦ عدم عبور ترافیک داده از AS

◦ مبدأ یا مقصد بسته ها

◦ مثال : ISP کوچک محلی

BGP پروتکلی است برای مسیریابی بین AS ها (OSPF) پروتکلی بود برای مسیریابی داخل یک AS .

: AS انواع

##### • Stub AS

◦ فقط یک راه ارتباطی به سایر AS ها دارد یعنی راه عبور بسته ها بین AS های مختلف نیست.

◦ ترافیک داده از AS عبور نمیکند یعنی از یک AS به AS دیگر از طریق این AS نمیتواند عبور کند چون گفته شد فقط یک ارتباط دارد.

- مبدا و مقصد بسته هاست.
- ISP های کوچک محلی معمولاً اینطوری هستند چون فقط یک اتصال دارند به ISP های دیگر برای برقراری اینترنت و اتصال دومی ندارند.

## BGP

### • **BGP : مسیریابی بین AS ها**

#### • **:AS انواع**

##### • **: Multihomed AS**

- بیش از یک راه ارتباطی به سایر AS ها
- عدم عبور ترافیک داده از AS
- مبدا یا مقصد بسته ها
- مثال : یک شرکت بزرگ که اجازه عبور ترافیک را نمی دهد

##### • **: Multi Homed AS**

- بیش از یک راه ارتباطی به سایر AS ها دارند.
- ترافیک داده ها را عبور نمیدهند.
- مبدا و مقصد بسته ها هستند.
- مثل شرکت بزرگی که چندین کانکشن به AS های دیگر دارد ولی اجازه عبور ترافیک را نمیدهد.

## **AS : مسیریابی بین AS ها**

### **• انواع AS**

#### **• Transit AS**

◦ بیش از یک راه ارتباطی به سایر AS ها

◦ عبور ترافیک داده از AS

◦ مثال : ISP های ملی یا بین المللی

#### **• Transit AS**

◦ بیش از یک راه ارتباطی به سایر AS ها دارد.

◦ اجازه عبور ترافیک را میدهد و از این AS ها میتوان برای عبور ترافیک AS های دیگر استفاده کرد و برای مسیریابی و رسیدن به یک AS خاص در مقصد میتوانند مورد استفاده قرار بگیرند.

◦ مثل ISP های ملی یا بین المللی

**• جلسات BGP**

• تبادل اطلاعات مسیریابی در یک جلسه

• جلسه: اتصال بین دو مسیریاب

• استفاده از **TCP**

• اتصالات طولانی هستند : **semipermanent connections**

از پروتکل TCP استفاده میشود. اتصالاتی که برقرار میشود بین روترهای مسیریابی جا میشود.

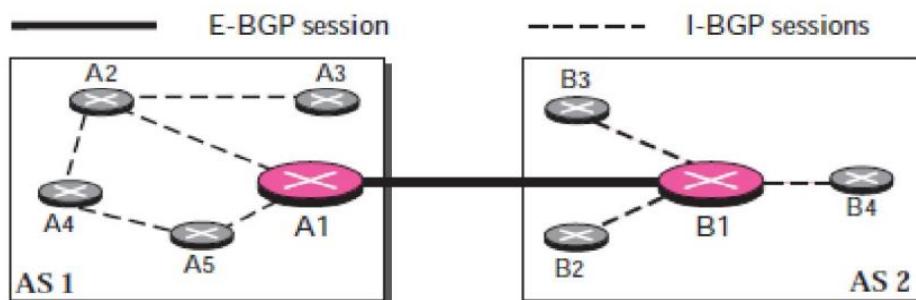
BGP Session شامل Session ها اطلاعات مسیریابی جا میشود.

## بیرونی و درونی BGP •

• دو نوع جلسه:

**AS بیرونی BGP (E-BGP)**: تبادل اطلاعات بین دو مسیریاب در دو AS •

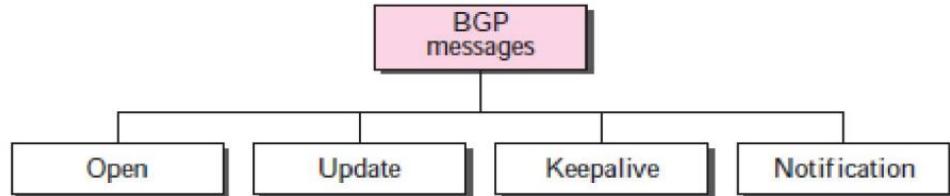
**AS درونی BGP (I-BGP)**: تبادل اطلاعات بین دو مسیریاب در یک AS •



درست است که این مسیریابها داخل یک AS قرار دارند ولی اطلاعاتی که ردوبدل میشود مربوط به BGP است یعنی مربوط به یک مسیریابی بین AS است.

در تصویر بالا ارتباطی که بین  $A^1$  و  $B^1$  است از نوع E-BGP است و بقیه ارتباطات که با خط چین نمایش داده شده است I-BGP است.

## ۰ انواع بسته (پیام)



• **Open :** ایجاد یک ارتباط (یا درواقع Session) .

• **Update :** به روز رسانی اطلاعات .

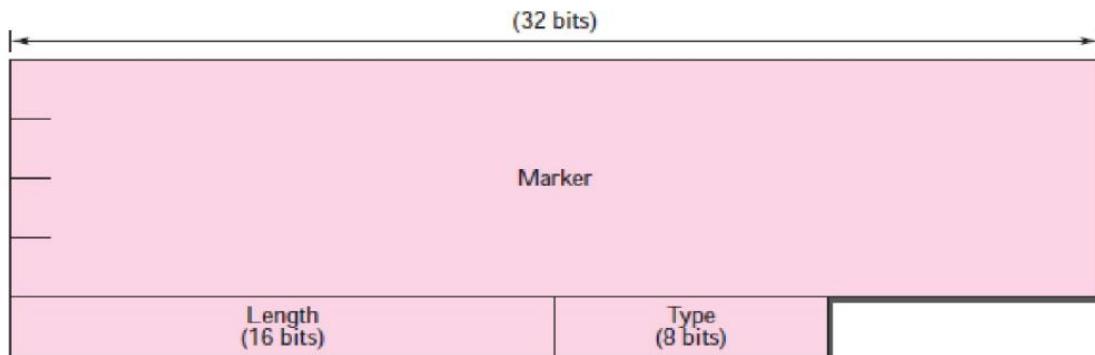
• **Keepalive :** باز نگه داشتن ارتباط زمانی که پیام update وجود ندارد .

• **Notification :** اعلام خطا زمانی که یک update خراب دریافت شده، یا زمان بستن ارتباط

توضیح TCP : Keepalive ها به گونه ای است که اگر مدتی پیامی ردوبدل نشود کانکشن ها را می بندند برای باز ماندن ان ارتباط نیاز است که پیام ردوبدل شود و زمانی که پیام آپدیتی وجود ندارد از Keepalive استفاده میکنیم که کانکشن بسته نشود.

## • قالب پیامها

### • سرآیند بسته های BGP

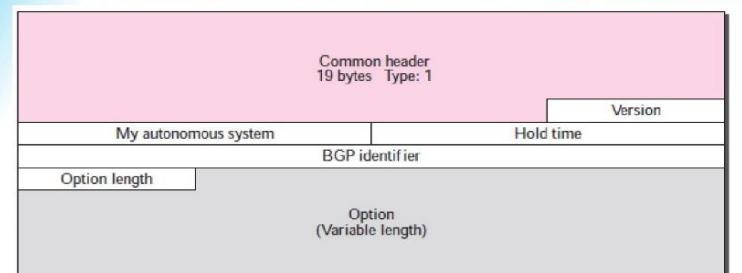


احراز هويت : Marker •

طول كل پيام : Length •

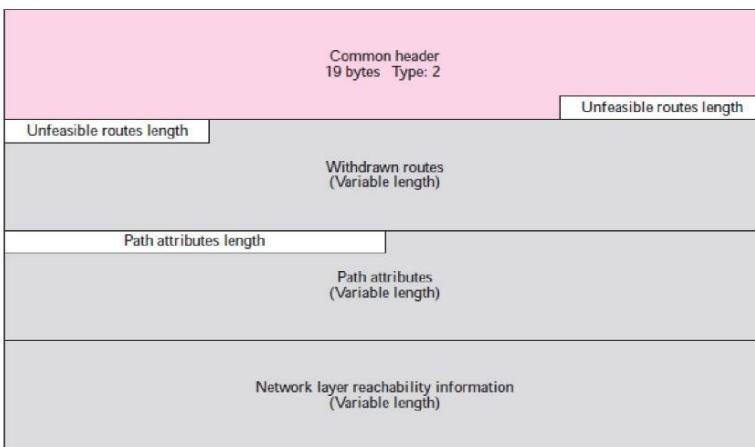
نوع پيام : Type •

## BGP



• قالب پیامها

**Open** •



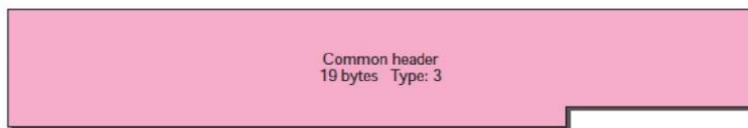
• Update •

11 of 34

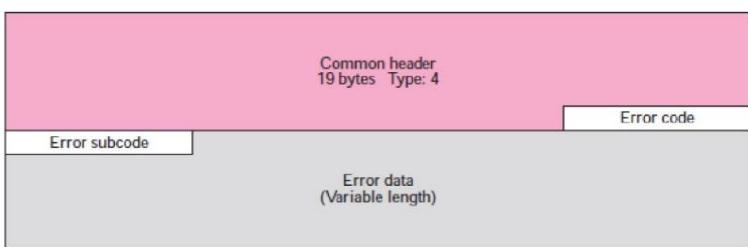
## BGP

• قالب پیامها

**Keepalive** •



**Notification** •



وارد جزئیات این قسمتها نمیشویم ولی شما کامل بخوانید و اطلاعات بیشتری از اینترنت سرچ کنید و بررسی کنید ببینید چه اتفاقی در پروتکل BGP میافتد و پیامها چگونه ردوبدل میشود و چه نتایجی دارند.

## BGP

### BGP •

• سه پروسه پروتکل:

• **: ایجاد رابطه همسایگی Neighbour acquisition**

• **: حفظ رابطه همسایگی Neighbour Reachability**

• **: بررسی در دسترس بودن شبکه ها Network reachability**

در مورد پروتکلهای BGP هم اطلاعات بیشتری میتوانید از اینترنت دریافت کنید.

## پروتکل های مسیریابی چندپخشی

چندپخشی در یک شبکه فیزیکی

چندپخشی در بین چند شبکه

الگوریتمهای چندپخشی

DVMRP

انواع ایجاد درخت تحویل چندپخشی

MOSPF

CBT

PIM

PIM-DM

PIM-SM

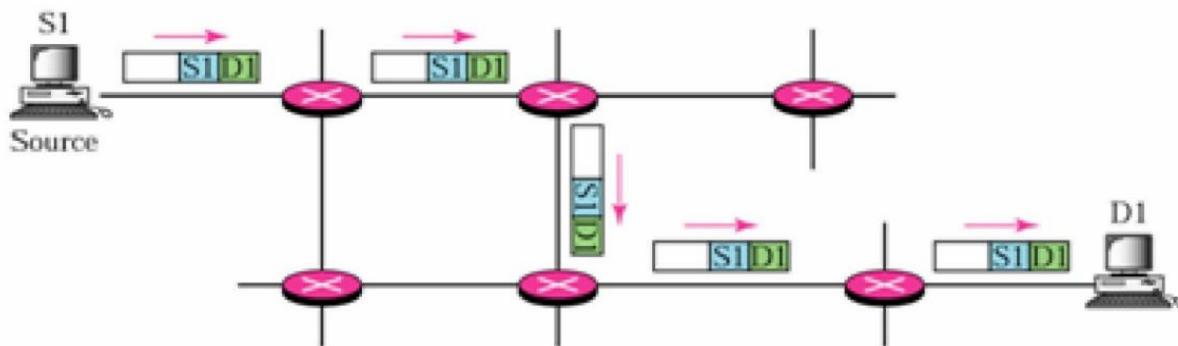
MBONE

مطالبی که درمورد مسیریابی Multi Cast خواهیم پرداخت به صورت تیترووار در این اسلاید  
امده است.

## پروتکل های مسیریابی چندپخشی

### ۰ انواع روش‌های ارسال بسته

#### ۰ تک پخشی (unicast) : ارسال بسته به یک مقصد مشخص

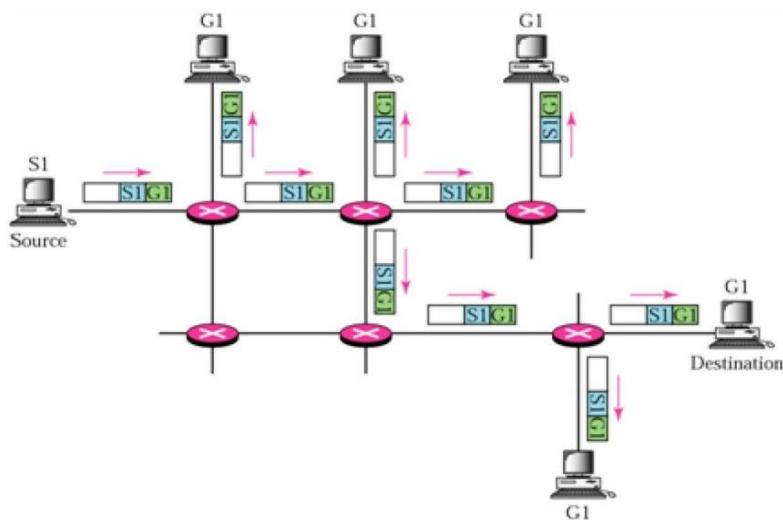


: یعنی بسته از Source مشخصی فرستاده میشود.

## پروتکل های مسیریابی چندپخشی

### ۰ انواع روش‌های ارسال بسته

#### ۰ چندپخشی (multicast) : ارسال بسته به گروهی از دریافت کننده‌ها

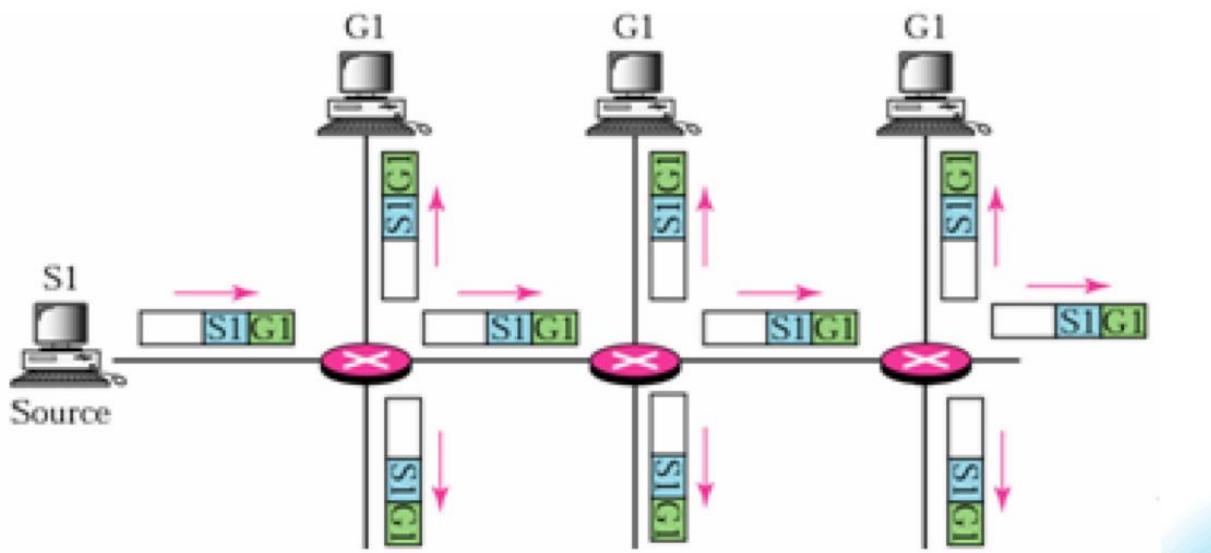


Multicast: یعنی یک Source مشخص به گروهی از Destination ها بسته ای را ارسال میکند. در تصویر بالا محتوای بسته ای که فرستاده شده برای همه یکسان است و میخواهد این بسته را به Host های منتخب بفرستد.

## پروتکل های مسیریابی چندپخشی

### ۰ انواع روش‌های ارسال بسته

#### ۰ همه پخشی (broadcast) : ارسال بسته به همه دریافت کننده ها



Broadcast: یعنی بسته باید به همه Host ها ارسال شود و استثنای ندارد.

## پروتکل های مسیریابی چندپخشی

### ۰ تعریف چندپخشی

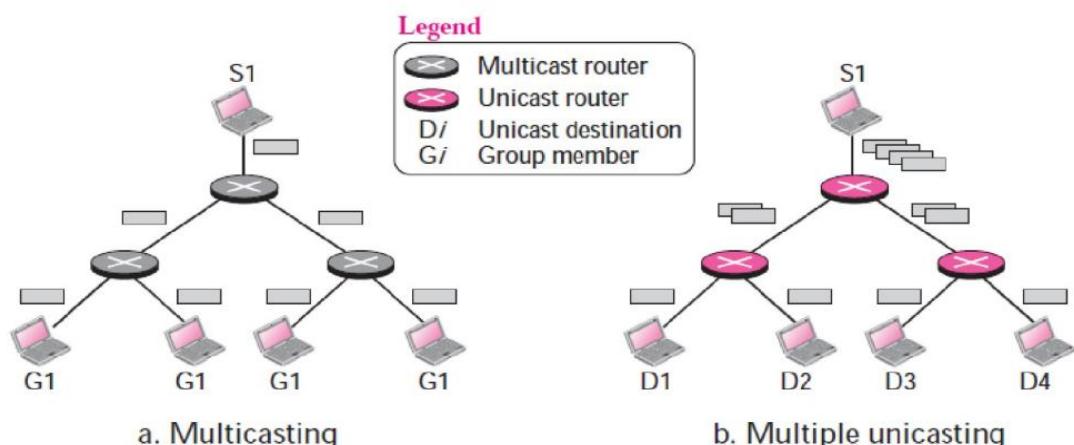
- چندپخشی IP (IP multicasting) یک فناوری حفظ کننده پهنه ای باند است که ترافیک را با رساندن همزمان یک دنباله از اطلاعات به هزاران دریافت کننده کاهش می دهد.

پس تعریف Multicasting : ارسال بسته از یک مبدا به چند مقصد و نه به تمام Host ها

IP Multicasting یک روش و فناوری است برای انجام Multicasting (یعنی یک بسته مشخص به چند هاب ارسال شود) و معمولاً دیتاایی که ارسال میشود Stream است. و طوری این اتفاق میافتد که کمترین پهنه ای باند ممکن مصرف شود.

## پروتکل های مسیریابی چندپخشی

### ۰ چندپخشی یا چند تک پخشی؟



چندپخشی به دو روش کلی انجام میشود:

فرض کنید ۱ Source میخواهد به ۴ عدد Destination ارسال کند، ۱ S1 میتواند ۴ تا کانکشن برقرار کند یعنی با هر Destination یک کانکشن برقرار کند و بسته را ارسال کند.

راه بهتر این است که S1 یکبار بسته را ارسال کند و روتراها این بسته را به سمت Destination های چندگانه تکرار کند.

در واقع روتراهای که در شکل B نمایش داده شده است روتراهای Unicast هستند و قابلیت Multicasting ندارند. ولی روتراهای شکل A قابلیت Multicasting دارند و این تفاوت بین این دو پیاده سازی است.

## پروتکل های مسیریابی چندپخشی

• چندپخشی یا چند تک پخشی؟

• دو اشکال عمدۀ چندپخشی بوسیله چند تک پخشی:

• چندپخشی کارایی بیشتری دارد

• اگر گیرنده ها زیاد باشند تاخیر بین بسته ها نمی تواند رعایت شود

فرض کنید Stream ای از داده داریم به این صورت که Packet اول که فرستاده شد بعد از حداقل ۵۰ میلی ثانیه پکت دوم باید فرستاده شود و همینطور در گیرنده بعد از ۵۰ میلی ثانیه از دریافت پکت اول باید پکت دوم دریافت شود. (Link Rate در فرستنده یک عدد مشخص است) اگر فرض اعداد گیرنده ها ۵۰۰ تا باشد ۵۰۰ بار باید بسته اول تکرار شود تا نوبت بسته دوم شود

ممکن است بعد از ۵۰۰ بار ارسال، زمان از ۵۰ میلی ثانیه خیلی بیشتر سپری شده باشد و بسته دوم نتواند سروقت ارسال شود حالا جدای از اینکه شبکه هم تاخیر دارد.

معمولا Streaming برای استفاده میشود مثلا یک ویدئو میخواهد توسط سورسی به گروهی از دریافت کننده ها فرستاده شود و اگر Unicast فرستاده شود آن ویدئو استریم به درستی نمیتواند ارسال شود.

## پروتکل های مسیریابی چندپخشی

- چرا چندپخشی؟
- چند پخشی، یک نیاز، حداقل در برخی سناریو ها
- سناریوهایی که اطلاعات (زیاد)ی به تعدادی (نه همه) میزبان ها در اینترنت باید ارسال شوند
- یک مثال عمومی: توزیع صوت یا ویدیو به میزبان هایی که به کنفرانس ملحق شده اند

### • مزایا:

- بکارگیری بهتر از پهنای باند

- پردازش سریعتر

- سیکل های مسیریابی مورد نیاز کمتر

چندپخشی در بعضی از سناریوها لازم است بعضی اوقات لازم است داده به چند هاست ارسال شود نه به همه. پس در سناریوهایی که قرار است اطلاعات معمولاً زیادی به تعدادی نه همه میزبانها در اینترنت باید ارسال شود Multicasting تنها گزینه است.

## پروتکل های مسیریابی چندپخشی

- کاربردهای چندپخشی
- هر کاربردی با چند گیرنده : یک به چند یا چند به چند
- توزیع ویدیوی زنده
- تحویل داده دوره ای - فناوری **push**
- مظنه بازار، نتایج ورزشی، مجلات، روزنامه
- تبلیغات
- تکثیر سرور و وب سایت
- دسترسی به پایگاه داده توزیع شده
- شبیه سازی محاوره ای توزیع شده (**DIS**)
  - بازی ها
  - واقعیت مجازی

یک مثال مالتی کستینگ Push آپدیت است.

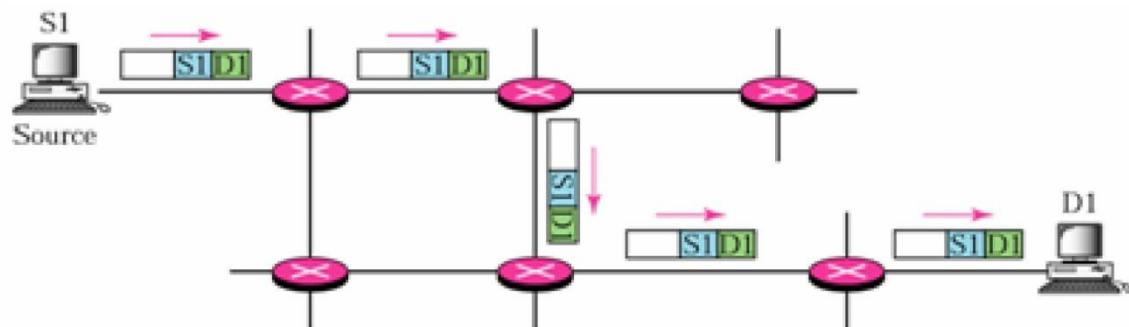
اسلاید مطالعه شود.

## جلسه هشتم:

### پروتکل های مسیریابی چندپخشی

#### ۰ انواع روش‌های ارسال بسته

##### ۰ تک پخشی (unicast) : ارسال بسته به یک مقصد مشخص

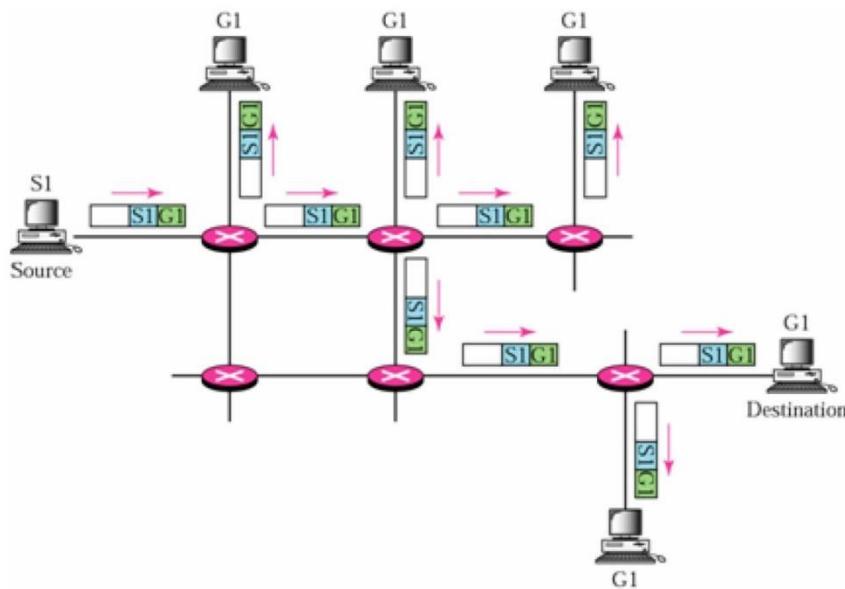


در خصوص روش‌های ارسال بسته گفتیم اینکه تک بخشی یعنی از یک سورس مشخص به یک مقصد مشخص می‌خواهیم پکت ارسال کنیم که = Unicast

# پروتکل های مسیریابی چندپخشی

## ۰ انواع روش‌های ارسال بسته

- ۰ چندپخشی (multicast) : ارسال بسته به گروهی از دریافت کننده‌ها



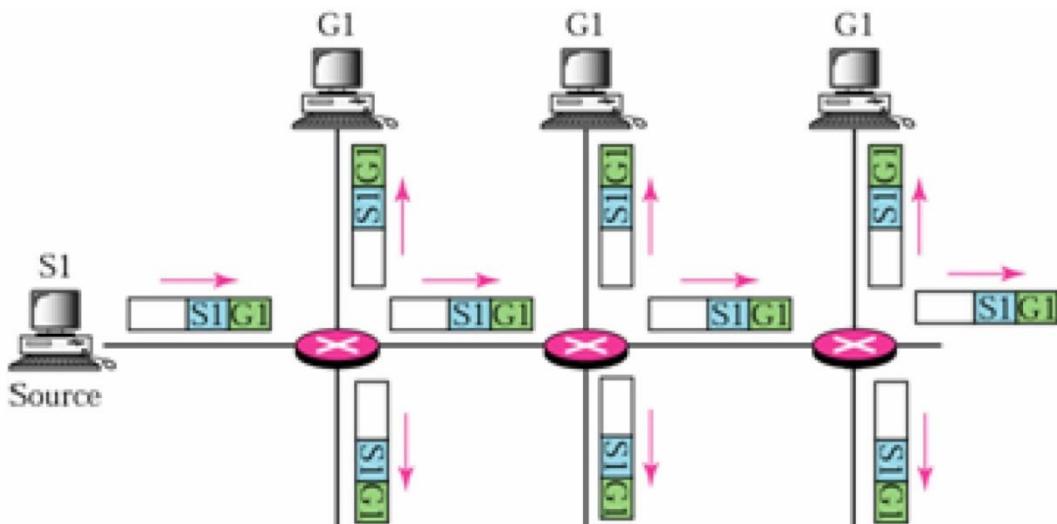
17 of 34

اگر بخواهیم یک بسته را به چند گیرنده بدهیم = Multicast است .

## پروتکل های مسیریابی چندپخشی

### ۰ انواع روش‌های ارسال بسته

#### ۰ همه پخشی (broadcast) : ارسال بسته به همه دریافت کننده ها



18 of 34

اگر بخواهیم به همه نود ها یا هاست ها بسته را ارسال کنیم می شود = Broadcast یا همه پخشی است.

# پروتکل های مسیریابی چندپخشی

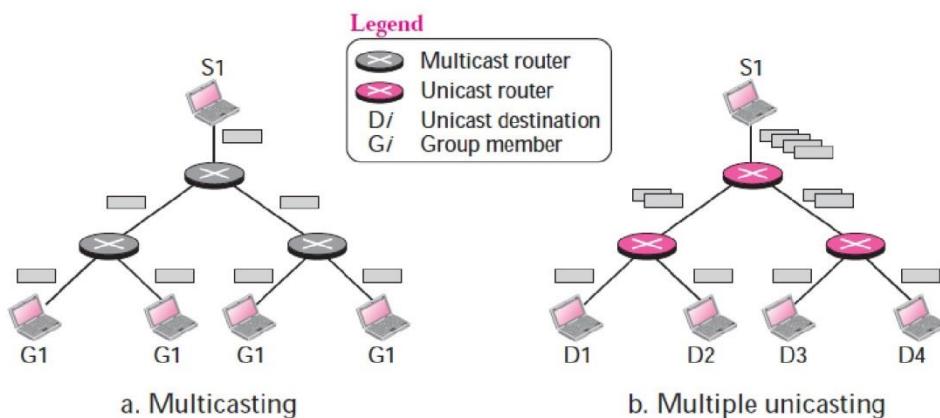
## ۰ تعریف چندپخشی

- چندپخشی IP multicasting (IP multicasting) یک فناوری حفظ کننده پهنای باند است که ترافیک را با رساندن همزمان یک دنباله از اطلاعات به هزاران دریافت کننده کاهش می دهد.

تکنولوژی است که پهنای باند را حفظ می کند در مقایسه با مولتی کست ، این کار را انجام می دهد راجع بهش صحبت خواهیم کرد.

# پروتکل های مسیریابی چندپخشی

## ۰ چندپخشی یا چند تک پخشی؟



Multipeling Cast را در مقابل Multicast می بینیم .

که در Multipeling Cast ما اگر فرض کنید به چهارتا هاست بخواهد بفرستیم چهار بار سورس باید دیتا را بفرستد برای همین پهنانی باند بیشتری مصرف می کند ولی در Multicast یک بار این سورس بسته را می فرستد و بعد روتراها این را به مسیرهایی که مقصد این بسته می توانند باشد فوروارد می کند یا به پورت هایی که مقصد ها روی آنها تعریف شده و وجود دارد یا از طریق آنها در دسترس هست فوروارد می کند و این باعث می شود که تعداد کمتری بسته در داخل شبکه وارد شده و پهنانی باند کمتری هدر برود.

## پروتکل های مسیریابی چندپخشی

• چندپخشی یا چند تک پخشی؟

• دو اشکال عمدۀ چندپخشی بوسیله چند تک پخشی:

• چندپخشی کارایی بیشتری دارد

• اگر گیرنده ها زیاد باشند تاخیر بین بسته ها نمی تواند رعایت شود

تفاوت و مقایسه چند پخشی با تک پخشی ؟

در واقع ما می توانیم Multi Unicast را با Multi Unicast انجام بدھیم یعنی فرض کنید به ۱۰ نفر می خواهید بسته ارسال کنید. ده بار ارسال کن.

دو تا اشکال وجود دارد از Multicasting به وسیله Multi Unicasting

- ۱- در چند پخشی یا Multicasting کارایی بیشتر هست
- ۲- اگر تعداد گیرنده ها زیاد باشد تاخیر بین بسته ها نمی تواند رعایت شود (که قبل از صحبت شد)

## پروتکل های مسیریابی چندپخشی

- چرا چندپخشی؟
- چندپخشی، یک نیاز، حداقل در برخی سناریو ها
- سناریوهایی که اطلاعات (زیاد)ی به تعدادی (نه همه) میزبان ها در اینترنت باید ارسال شوند
- یک مثال عمومی: توزیع صوت یا ویدیو به میزبان هایی که به کنفرانس ملحق شده اند

### • مزایا:

- بکارگیری بهتر از پهنای باند
- پردازش سریعتر
- سیکل های مسیریابی مورد نیاز کمتر

22 of 34

به طور کلی چرا Multicasting یا چند پخشی نیاز هست ؟

در بخش از سناریوها باید Multicasting انجام بگیرد.

## پروتکل های مسیریابی چندپخشی

- کاربردهای چندپخشی
- هر کاربردی با چند گیرنده : یک به چند یا چند به چند
- توزیع ویدیوی زنده
- تحویل داده دوره ای - فناوری **push**
- مظنه بازار، نتایج ورزشی، مجلات، روزنامه
- تبلیغات
- تکثیر سرور و وب سایت
- دسترسی به پایگاه داده توزیع شده
- شبیه سازی محاوره ای توزیع شده (**DIS**)
  - بازی ها
  - واقعیت مجازی

23 of 34

مثال هایی از اون رو هم در این اسلاید می توانید که ببینید.

مثل ویدیو زنده فرض کنید همین کلاس هایی که به صورت آنلاین داریم و خیلی از کاربردهایی هست که از یک مبدأ به چندین مقصد نه همه هاست ها بلکه به تعدادی از هاست ها در شبکه ارسال شود و این همان کاربرد Multicasting هست.

## پروتکل های مسیر یابی چند پخشی

### ۰ چند پخشی

- ۰ ارسال به یک آدرس شبکه چند پخشی (multicast IP)
- ۰ چند پخشی از کلاس D آدرس شبکه استفاده می کند
- ۰ هر آدرس چند پخشی ≈ تعدادی میزبان (گروه میزبان)
- ۰ ارسال کننده می تواند عضو گروه نباشد
- ۰ امنیت کمی وجود دارد
- ۰ قابلیت اطمینان وجود ندارد

در Multicasting آدرس مقصد یک آدرس چند بخش خواهد بود یا آدرس خواهد بود که بهش Multicast Ip گفته می شود.

در Multicasting از کلاس D آدرس شبکه استفاده می شود هر آدرس Multicast برابر با تعداد هاست هایی هست که بسته قراره به آنها ارسال شود، این میزبان هایی که آدرس Multicast را دارند بهش گروه گفته می شود که ارسال کننده می تواند که عضو گروه نباشد یعنی لزومی ندارد که چه کسی می خواهد به یک گروه خاصی دیتا بدهد خودش عضو گروه باشد امنیت Multicasting کم هست و چون بسته دارد به چندین هاست ارسال می شود و می تواند هاست مهاجم هم وجود داشته باشد.

قابلیت اطمینان هم وجود ندارد یعنی بسته واقعاً به درستی و به ترتیب به همه هاست ها رسیده باشند چون مثل Tcp امکان Ack وجود ندارد مثلاً ۱۰ هزار تا هاست گیرنده داریم و یک پکت می خواهیم بفرستیم که ۱۰ هزار تا Akc بگیریم اصلاً درست نیست.

# پروتکل های مسیریابی چندپخشی

## ۰ آدرس های چندپخشی

۰ دامنه آدرس : **239.255.255.255 تا 224.0.0.0**

CIDR	Range	Assignment
224.0.0.0/24	224.0.0.0 → 224.0.0.255	Local Network Control Block
224.0.1.0/24	224.0.1.0 → 224.0.1.255	Internet Control Block
	224.0.2.0 → 224.0.255.255	AD HOC Block
224.1.0.0/16	224.1.0.0 → 224.1.255.255	ST Multicast Group Block
224.2.0.0/16	224.2.0.0 → 224.2.255.255	SDP/SAP Block
	224.3.0.0 → 231.255.255.255	Reserved
232.0.0.0/8	232.0.0.0 → 224.255.255.255	Source Specific Multicast (SSM)
233.0.0.0/8	233.0.0.0 → 233.255.255.255	GLOP Block
	234.0.0.0 → 238.255.255.255	Reserved
239.0.0.0/8	239.0.0.0 → 239.255.255.255	Administratively Scoped Block

25 of 34

آدرس Multicasting ، Domain به این صورت هست که از ۲۲۴.۰.۰.۰ تا ۲۳۹.۲۵۵.۲۵۵.۲۵۵ به عنوان مثال تقسیم بندی این محدوده‌ی آدرس را در جدول می‌بینیم. مثلاً ۲۲۴.۰.۰.۰/۲۴ معادل Cidr آدرس هست. که رنج ۲۲۴.۰.۰.۰/۲۲۴.۰.۰.۰.۲۵۵ هست یعنی ۸ بیت کم ارزش آدرس هاست هست که برای این Local Network هستش یعنی مخصوص این هست و دسته بعدی برای Internet.

## پروتکل های مسیریابی چندپخشی

### ۰ آدرس های چندپخشی

۲۲۴.۰.۰.۰ تا ۲۲۴.۰.۰.۲۵۵ تخصیص یافته برای پروتکل شبکه

Address	Assignment
224.0.0.0	Base address (reserved)
224.0.0.1	All systems (hosts or routers) on this network
224.0.0.2	All routers on this network
224.0.0.4	DMVRP routers
224.0.0.5	OSPF routers
224.0.0.7	ST (stream) routers
224.0.0.8	ST (stream) hosts
224.0.0.9	RIP2 routers
224.0.0.10	IGRP routers
224.0.0.11	Mobile Agents
224.0.0.12	DHCP servers
224.0.0.13	PIM routers
224.0.0.14	RSVP encapsulation
224.0.0.15	CBT routers
224.0.0.22	ICMPv3

26 of 34

این رنج در داخل اسلاید برای پروتکل های شبکه تخصیص پیدا کرده است که طبق جدول در داخل اسلاید مشخص کرده به چه صورت هست.

= ۲۲۴.۰.۰.۱ دلالت دارد به همه هاست های در Network Broad Casting یعنی انجام می شود یعنی اگر آدرس مقصد یک پتک باشد یعنی ۲۲۴.۰.۰.۱ باشد این پکت را همه های هاست ها در داخل شبکه برمی دارند و پردازش اش می کنند و روتر ها هم باید بسته ها را داخل شبکه فوروارد کنند.

= ۲۲۴.۰.۰.۲ همه روترها روی این Network یعنی روترها این بسته را پردازش خواهند کرد.

= ۲۲۴.۰.۰.۴ مخصوص Dmvrp روتر ها هست (روتر هایی که این پروتکل را دارند)

و الی آخر در داخل جدول قابل نمایش هستند که هر کدام آدرس یک Broadcast را مشخص میکند.

## پروتکل های مسیریابی چندپخشی

### • چندپخشی

◦ دو نوع گروه میزبان، ثابت و موقت

◦ گروه ثابت

◦ گروه ثابت است اما عضویت اعضا ثابت نیست

◦ ممکن است گروه خالی از عضو باشد

◦ از DNS استفاده می شود

### ◦ گروه موقت:

◦ به صورت پویا ایجاد می شود

◦ بدون عضو متوقف می شود

27 of 34

در Multicast دو نوع گروه میزبان داریم یعنی Host Group ها دو نوع هستند.

به صورت کلی یا ثابت هستند یا موقت(یعنی حتما باید عضو بشوند و بعد از عضویت خارج بشوند)

گروه ثابت یعنی یک تعریف مشخصی دارد -منظور این نیست که اعضا ثابت هستند تعریفش ثابت و مشخص است گروه ثابت هستند اما عضویت اعضا ثابت نیست. ممکن است که گروه خالی از عناصر باشد یعنی همچین تعریفی وجود نداشته باشد.

از Dns هم استفاده میشود

گروه موقت به صورت پویا ایجاد می‌شود یعنی یک هاست می‌تواند گروه ایجاد کند و هاست‌های دیگر را به گروه دعوت کند برای اینکه Multicasting انجام بشود و گروهی که عضو نداشته باشد تعریف نمی‌شود و دیگر گروه نیست و متوقف می‌شود.

## چندپخشی در یک شبکه فیزیکی

- تعیین آدرس شبکه چندپخشی توسط پروسه فرستنده
- انطباق آدرس شبکه به آدرس‌های فیزیکی متناظر و ارسال بسته به آن آدرس‌ها
- قابل استفاده نیست ARP
- وابستگی نحوه نگاشت آدرس IP به آدرس فیزیکی به پشتیبانی لایه پیوند داده از آدرس‌های چندپخشی فیزیکی

چند پخش شده در یک شبکه فیزیکی

آدرس شبکه چندپخشی توسط پروسه فرستنده تعریف می‌شود یعنی Sender Process آدرس شبکه مشخص می‌کند.

یک مپینگ بین آدرس شبکه و آدرس‌های فیزیکی متناظر انجام می‌شود و در این شرایط Arp هم قابل استفاده نیست چون Arp مخصوص آدرس‌های Unicast بود.

نحوه و روش مپینگ IP آدرس به فیزیکال آدرس یا همان آدرس مک به پشتیبانی لایه ۲ پیوند داده از آدرس‌های چندپخشی و فیزیکی وابسته هست یعنی چه پشتیبانی می‌کند به این بستگی دارد که چگونه نگاشت انجام می‌شود.

## چندپخشی در یک شبکه فیزیکی

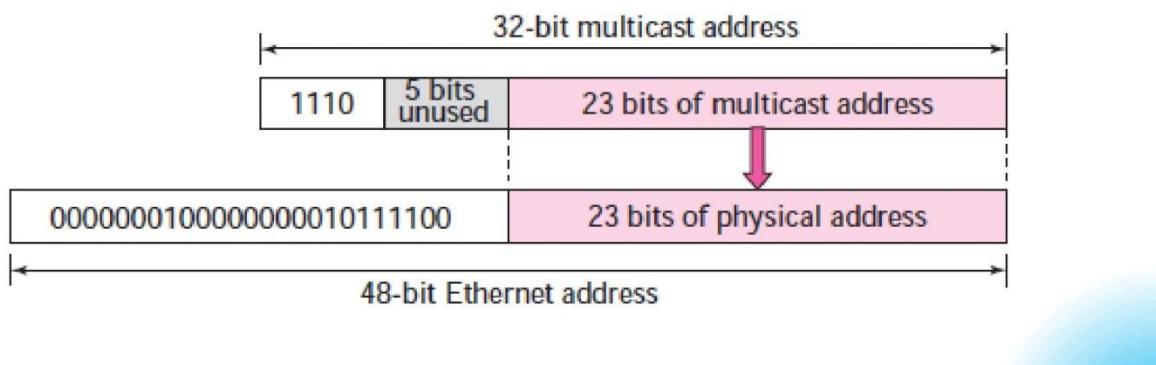
۰ پشتیبانی از آدرس چندپخشی فیزیکی :

۰ آدرس فیزیکی : ۴۸ بیت

۰ ۲۵ بیت اول = **00000001 00000000 01011110**

آدرس چندپخشی

۰ ۲۳ بیت باقیمانده : گروه



اگر از آدرس چند بخشی فیزیکی پشتیبانی بشود همان طور که می دانید آدرس فیزیکی یا مک آدرس ۴۸ بیتی هست اول آن در آدرس چند بخشی فیزیکی همانند داخل اسلايد هست یعنی آدرس های فیزیکی پیدا نمی کنند به صورت Unicast باشد و ۲۵ بیت اول آن پترن اش است به این صورت می باشد و این ۲۵ بیت آدرس چند پخشی هست و ۲۳ بیت باقیمانده (چون کلا ۴۸ بیت است).

هم گروه هست - این ۲۳ بیت از ۳۲ بیت آدرس به چه صورت پر میشود.

فرض کنید آدرس Multicast هست که داریم . می خواهیم به این آدرس بسته ارسال ارسال کنیم و می خواهیم به آدرس فیزیکی در شبکه مپ و بسته را ارسال کنیم و هاست ها دریافت کنند. ۲۵ اول که این الگو را دارد ۲۳ بیت باقیمانده که می شود ۲۳ کم ارزش Multicast آدرس و ۴ بیت هم که کلاس را مشخص میکند ۱۱۱۰ و ۲۳ بیت هم که داریم که می شود ۲۷ بیت از

۳۲ بیت باقی می‌ماند که استفاده نمی‌شود یعنی ۲ به توان ۵ آدرس مختلف را به یک آدرس فیزیکی یا آدرس مک مپ می‌کنیم یک همچنین شرایطی هست برای تبدیل Multicast Ip به آدرس فیزیکی

## چندپخشی در یک شبکه فیزیکی

- ۰ استفاده از ۲۳ بیت سمت چپ آدرس شبکه برای نگاشت
- ۰ چشم پوشی از ۵ بیت آدرس (بیت‌های شماره ۲۳ تا ۲۷)
- ۰ ← نگاشت ۳۲ آدرس غیر یکتا در یک آدرس فیزیکی
- ۰ ← نیاز به فیلتر کردن بسته‌ها در درایور مقصد
- ۰ دو دلیل دیگر برای فیلتر کردن:
  - ۰ محدود بودن آدرس چندپخشی همزمان در برخی آداتورها، در صورت تجاوز از این محدوده همه بسته‌های چندپخشی را دریافت می‌کنند
  - ۰ استفاده از جدول درهم ساز در برخی آداتورها، در صورت یکسان بودن مقدار درهم ساز ۲ آدرس بسته‌های اضافی عبور می‌کنند
- ۰ با این وجود هنوز هم سربار کمی در میزبان‌ها یی که عضو گروه نیستند وجود دارد

از ۲۳ بیت سمت چپ آدرس برای نگاشت استفاده می‌کنیم از پنج بیت بعدی چشم پوشی می‌کنیم بنابراین نگاشت ۳۲ بیت آدرس غیر یکتا در یک آدرس فیزیکی خواهد بود یعنی ۳۲ آدرس Broad Cast مختلف را به یک آدرس فیزیکال مشترک Map می‌کنیم بنابراین این بسته‌ها در درایور گیرنده باید فیلتر شوند.

فرض کنید برای شبکه یک می‌خواهیم بفرستیم ولی ۳۱ شبکه دیگر هم از همین آدرس فیزیکال برای Broad Cast استفاده می‌کنند (منظور گروه می‌باشد) به یک گروه مشترک می‌خواهیم بسته ارسال کنیم ، ۳۱ گروه دیگر هم هستند که از همین فیزیکال آدرس استفاده می‌کند یعنی

آی پی آدرس Broad Cast شان مپ می شود به همین آدرس فیزیکال Broad Cast بنابراین درایور گیرنده باید یک فیلترینگ را انجام بدهد و بقیه اون ۳۱ گروه را به سمت لایه های بالاتر هدایت نکند و Drop کند.

دو دلیل دیگر برای فیلترینگ در اینجا نمایش داده شده است

۱- آدرس‌های چند بخش همزمان در برخی آداتورها محدود هستند در این صورت اگر از این محدوده تجاوز کند در این صورت همه بسته های چند بخشی را دریافت می‌کند.

۲- بعضی از نت ورک آداتورها از جدول هشینگ (درهم ساز) استفاده می‌کند و اگر مقدار هش شده یکسان باشد در آدرس بسته ها عبور می‌کند در صورتی که محور هش شده یکسانی در بیاید.

هاست هایی که عضو گروه نیستند و نباید این بسته ها را بگیرند ولی با توجه به فیلترینگ کمی هم که هست هنوز سربار کمی هم روی هاست ها وجود دارد و یک مقداری ممکنه بسته ها از فیلترشان عبور بکند در صورتی که نباید بکند.

## چندبخشی در یک شبکه فیزیکی

۰ مثال : آدرس چندبخشی 232.43.14.7 را به آدرس چندبخشی فیزیکی متاظر آن تبدیل کنید

آدرس فوق به آدرس چندبخشی تبدیل شود؟

سوال آفای آیت: یعنی مک را تغییر می‌دهد؟

جواب استاد: ما یک آدرس مک در هر کارت شبکه ای داریم که یونیک هست و مخصوص این کارت شبکه هست اما وقتی که بخواهیم بسته‌ای از یک نود به نود دیگری بفرستیم در اون بسته ای که ایجاد می‌کنیم باید آدرس مک خودمان که فرستنده هستیم و آدرس مک گیرنده را در داخل بسته قرار بدهد و این باید آدرس مک که اسمش را آدرس فیزیکال گذاشتیم آدرس مکی هست که در قسمت گیرنده‌ی بسته داریم قرارش می‌دهیم پس آدرس مک کسی تغییر نمی‌کند و این آدرس مک گیرنده آن بسته هست که الان باید تعیین شود Arp هم همین کار را می‌کند اگر می‌خواستیم یک بسته را به یک هاست ارسال کنیم اگر آدرس مک را نمی‌دانستیم با استفاده از Arp آدرس مک را پیدا می‌کردیم و بعد با این آدرس بسته را به هاست ارسال می‌کردیم الان هم می‌خواهیم یک بسته Multicast بفرستیم و باید آدرس مک گیرنده را در بسته ای که می‌خواهیم ارسال کنیم پر کنیم با استفاده از پروتکل Arp از هاست گیرنده بپرسیم که آدرس مک چیه؟ ولی در Multicast چون یک هاست نیست و چند نفر هستند بنابراین باید یک آدرس مک Multicast استفاده کنیم و آدرس مک Multicast است.

## چندپخشی در یک شبکه فیزیکی

• مثال : آدرس چندپخشی 232.43.14.7 را به آدرس چندپخشی فیزیکی متناظر آن تبدیل کنید

• پاسخ :

• ۲۳ بیت سمت راست آدرس را به مبنای ۱۶ تبدیل کنیم می‌شود:  
**2B:0E:07**

• آنرا به آدرس چندپخشی فیزیکی (01:00:5E:00:00:00) اضافه کنیم:

**01:00:5E:2B:0E:07 •**

۲۵ بیت سمت چپ به صورت زیر خواهد بود :

.....1.....1011100

و مپینگ رو هم که قبلاً توضیح دادیم .

آدرس باینری به صورت زیر می شود .

1110100001010110001110.....111

۴ بیت از سمت چپ برای کلاس D می رود (1110) و ۵ بیت هم استفاده نمی شود (10000)

این ۲۳ بیت را باید کنار اون ۲۵ بیت قرارش بدھیم .

.....1.....101110001010110001110.....111

این می شود آدرس فیزیکی متناظرش و به صورت زیر می باشد .

• ۱۰۰:۵e:۲b:۰e:۰۷

این می شود آدرس Multicast فیزیکال برای آدرس Ip

نحوه‌ی تبدیل به هگز :

هر ۸ بیت دورقم ۴ بیت در نظر میگیریم به عبارتی دوتا ۴ بیت خواهد بود که هر کدام یک عدد می شود که تبدیل به هگز را روی آن خواهیم داشت .

سوال آقای آیت :

What Is Subnetmask In This Simple ?

جواب استاد :

هاست ها هر کدامشان یک آدرس Ip دارند و به اون آدرس Ip کاری نداریم و یک آدرس Ip داریم مثل همین مثال این آدرس ای پی Multicastin ما برای یک گروه هست حالا چه کسانی عضو این گروه هستند هر کس که خودش را عضو گروه معرفی کرده باشد به چه صورت به روتر Multicast Enabled اعلام میکند که من میخواهم که عضو این گروه با این Ip بشوم وقتی که اعلام بکند در جدول مسیریابی اون روتر می آید که این Host عضو این گروه هست . و Subnetmask این جا معنی ندارد و یک آدرس ای پی Multicast داریم و برای Ip Unicast برای Subnetmask اینکه مشخص کنیم که Classfull هست جزو کدام؟ و اگر Class Less هست به چه صورت خواهد بود .

آدرس ای پی Multicast متعلق به یک گروه هست و به این صورت استفاده می شود .

## چندپخشی در یک شبکه فیزیکی

- مثال : آدرس چندپخشی 238.212.24.9 را به آدرس چندپخشی فیزیکی متناظر آن تبدیل کنید

خودتان انجام بدهید

جواب :

• ۱۰۰:۵۴:۱۸:۰۹

# پروتکل های مسیریابی

مسیریابی چندپخشی (Multicast)

چندپخشی در یک شبکه فیزیکی

چندپخشی در بین چند شبکه

مقدمه

مسیریابی حالت پیوند چندپخشی

**MOSPF**

مسیریابی بردار فاصله چندپخشی

**DVMRP**

**CBT**

چندپخشی مستقل از پروتکل (PIM)

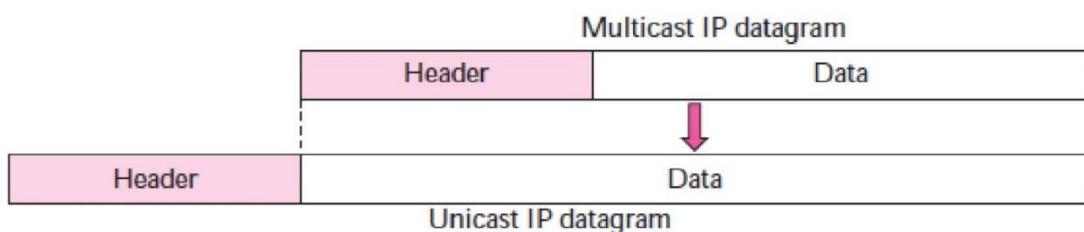
**MBONE**

3 of 37

## چندپخشی در یک شبکه فیزیکی

• عدم پشتیبانی از آدرس چندپخشی فیزیکی :

• استفاده از تونل زنی (**tunneling**)



اگر آدرس Multicast در شبکه پشتیبانی می‌شد یعنی توسط کارت های شبکه یعنی همان رویه قبلی که برای تبدیل آدرس آی‌پی Multicast و آدرس فیزیکال Multicast انجام می‌شود در نتیجه Multicasting انجام می‌شود ولی اگر از آدرس فیزیکال Multicast در شبکه پشتیبانی نشود یعنی کارت شبکه نداشته باشیم می‌توانیم از Tunneling استفاده کنیم با استفاده از تونل زنی می‌توان Multicast را انجام داد.

به وسیله Tunneling ip Diagram می‌ایم قرار می‌دهیم و ارسال می‌کنیم به تک تک هاست های شبکه این Unicast Datagram آخرین بحث Multicast در شبکه های فیزیکی بود.

## چندپخشی در بین چند شبکه

- ترافیک چندپخشی محدود به یک شبکه نیست
- مخاطراتی وجود دارد، از جمله ایجاد حلقه
- جهت پیشگیری از این مخاطرات: پروتکل های مسیریابی چندپخشی

اما Multicasting در بین چند شبکه را می‌خواهیم که ببینیم به چه شکلی است Multicasting محدود به یک شبکه نیست یعنی اینجور نیست که در یک شبکه محلی در داخل یک Lan بخواهیم دیتا را به چند هاست برسانیم مثلًاً در اپلیکیشن ها مخاطبین در داخل Local Network نیست و ممکنه مخاطبین یک ارتباط جاهای مختلف کشورهای متفاوتی باشند و قطعاً باید یک ترافیکی بین شبکه‌ها حرکت کند در Multicasting بین چند شبکه مشکلات و مخاطراتی که وجود دارد:

۱- ایجاد حلقه- احتمال ایجاد حلقه وجود دارد برای اینکه از این مخاطرات جلوگیری کنیم مشکلات بعدی که ممکن است وجود داشته باشد را از قبل پیش‌بینی داشته باشیم و راه حل داشته باشیم از پروتکل های Multicasting استفاده می‌کنیم.

میخواهیم راجع به پروتکل های Multicast صحبت کنیم

## چندپخشی در بین چند شبکه

### ۰ دو نیازمندی:

- ۰ تعیین مشترکین چندپخشی : RFC2236 ، IGMP
- ۰ تعیین حوزه چندپخشی : فیلد TTL برای این منظور استفاده می شود:
  - ۰ محدود به میزبان مبدأ : TTL=0
  - ۰ محدود به زیرشبکه : TTL=1
  - ۰ یا بیشتر : عملکرد مسیریاب به آدرس بستگی دارد: TTL=2
  - ۰ تک پرشی، حذف توسط مسیریاب 224.0.0.0 – 224.0.0.255
  - ۰ سایر : جلورانی توسط مسیریاب

دوتا نیازمندی اولیه و اصلی دارد :

۱- مشترکین Multicasting را باید بشناسیم یعنی اینکه برای این گروه چه هاست هایی وجود دارد و هاست ها عضو چه گروهی هستند را باید تعیین کنیم این یک نکته است پروتکلی که برای این کار استفاده می شود پروتکل Igap Internet Group هستش کار مدیریت گروه را در اینترنت انجام می دهد و رفرنس Management Protocol Rfc۲۲۳۶ این پروتکل را معرفی کرده است و توضیحات لازم را داده است.

- ۲ و همینطور حوزه چند بخشی در کجاها این چند بخشی انجام بگیرد فیلم  $Ttl$  بسته برای این منظور استفاده شده است که حوزه چند بخشی را مشخص کند مثلاً :

$Ttl=0$  : محدود به میزبان مبدا هست و بسته خارج نشود از Sending Host و در همان هاست بماند

$Ttl=1$  : در همان Subnet بماند و در شبکه دیگری نرود

$Ttl=2$  Or More : بستگی به آدرس مقصد دارد

۲۲۴.۰.۰.۲۵۵/۲۲۴.۰.۰.۰ عملکرد روتر در اینجا Singlehop هست یعنی مسیریابها آن را حذف می‌کند و عبورش نمی‌دهند در اولین گام باید هدایت شود به گام‌های بعدی ولی اگر آدرس های دیگری برای Multicast استفاده شود اون وقت مسیریابها هدایت می‌کند به پورت هایی که احساس می‌کند مقصدها آنجا قرار دارند ممکنه یک روتر به یک پورت خروجی ارسال کند و ممکنه روتربه ۵ تا پورت خروجی اش ارسال کند و این بستگی به وضعیت هاست‌های گیرنده دارد.

## چندپخشی در بین چند شبکه

- مسیریابی چندپخشی : پیدا کردن مسیر بهینه
- تشکیل درخت به کمک الگوریتم‌های ارسال چندپخشی و IGMP
- در تک پخشی : درخت کوتاه ترین مسیر
- در چندپخشی : دو نوع درخت
- درخت مبتنی بر مبدا (source based) : ترکیب مبدا و گروه
- درخت مشترک در گروه (group shared) : گروه

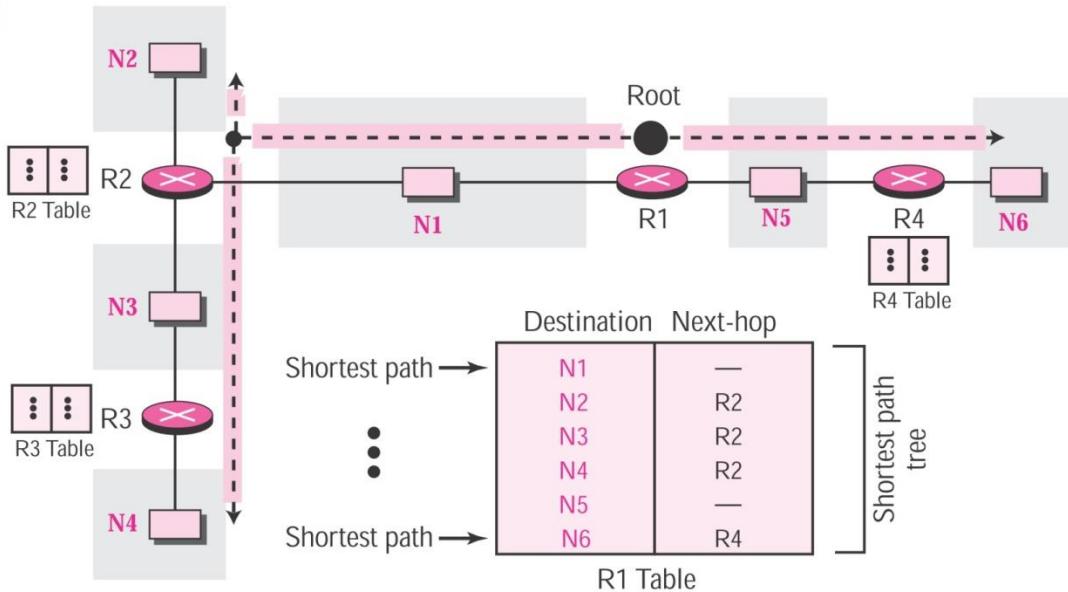
ما بعد از معرفی این که برای Multicast در شبکه نیاز به مسیریابی هم داریم چون به هر حال بسته باید به یک مقصد برود (که چندین وقت در اینجا خواهد بود) برای این چندین مقصد باید مسیر پیدا شود و مسیر بهینه باید پیدا شود.

در مسیریابی Multicast درخت تشخیص خواهد شد در تک بخشی درخت کوتاهترین مسیر تشکیل می‌شود از هر فرستنده به همه گیرنده‌ها ولی در چند پخشی ما دو نوع درخت داریم:

- ۱- درخت مبتنی بر مبدا Source Base Tree که در واقع ترکیبی از مبدا و گروه است به عبارتی مبدا درش مهم و تاثیر گذار هست
- ۲- درخت مشترک در گروه یا Group Shared درش تاثیر گذار هست و اون مبدا مهم نیست.

## چندپخشی در بین چند شبکه

### • درخت کوتاه‌ترین مسیر



درخت کوتاهترین مسیر را در این مثال می بینیم که در Unicast استفاده می شود فرض کنید که روتر  $R_1$  ما باشد در این صورت بقیه نود ها را می بینیم که به چه شکل درخت برای شان تشکیل شده است.

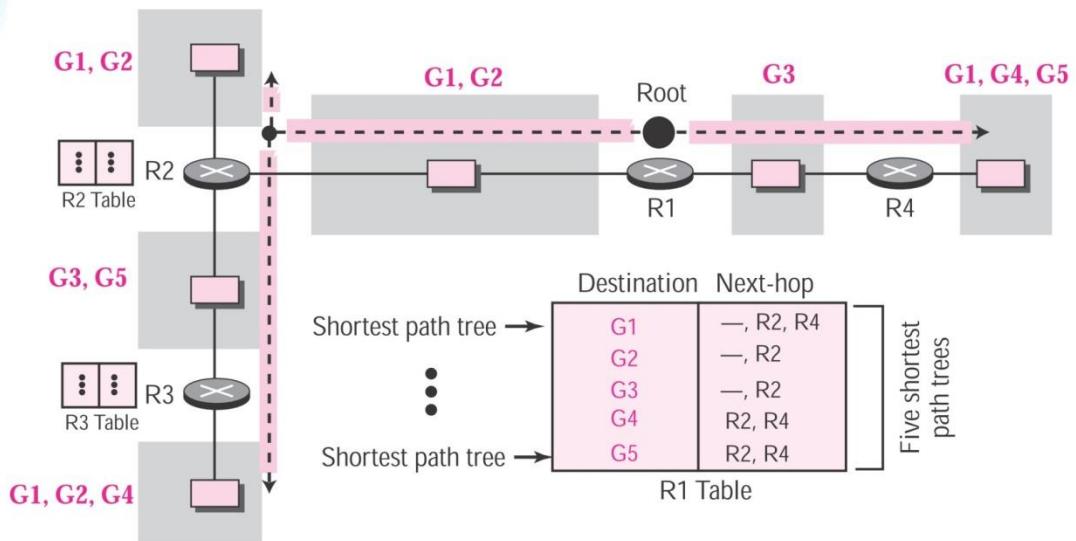
نداریم چون اینها شبکه هایی هستند که مستقیم  $R_1$  به  $N_1, N_5$  - Next Hop مثلاً اینجا از  $R_1$  می شود وصل هستند ولی  $N_2$  - Next Hop.

یعنی روتر  $R_1$  بسته را باید به  $R_2$  بدهد که اون بفرستد برای  $N_2$  یا مثلاً به  $N_6$  بخواهیم که برویم  $R_4$  - Next Hop هست و اینجا درخت تشکیل شده با ریشه  $R_1$  بهش درخت کوتاهترین مسیر گفته می شود این جدول مسیریابی  $R_1$  هست که با تشکیل درخت کوتاهترین مسیر می تواند که پرس کند و بعد از این می تواند که استفاده کند برای فوروارد کردن بسته ها سمت هر کدام از مقصد ها در این شکل ما یکسری نتورک داریم که در هر کدام از آنها یک تعدادی هاست داریم و هر کدام از هاست ها یک آدرس Unicast دارند و سورس .

ما فرض کنید یکی از هاست های  $N_5$  بخواهد بسته ای را ارسال بکند به یکی از هاست های  $N_2$  و بسته اش را می دهد به روتر  $R_1$  و  $R_1$  از جدول مسیریابی نگاه می کند و می بیند که مقصد هاست کجاست که بسته می خواهد به آنجا برود در واقع سمت روتر  $R_2$  هست در Multicast به همین سادگی هست ولی در Unicast اسلاید بعدی توضیح می دهیم.

## چندپخشی در بین چند شبکه

### ۰ درخت مبتنی بر مبدأ (source based)



9 of 37

در شبکه N<sup>۲</sup> Multicast یکسری هاست هستند که این هاست ها بعضی هایشان متعلق به گروه G<sup>۱</sup> هست و بعضی هم متعلق به گروه G<sup>۲</sup> هستند و ممکن است اینجا باشد و هم در گروه G<sup>۱</sup> عضو باشد و هم در گروه G<sup>۲</sup> ولی هر دو گروه در آن جا یک عضوی دارند.

کل<sup>ا</sup> ۵ گروه داریم این شکل نشان می‌دهد که اعضای گروه ها به چه شکلی در نتورک قرار گرفته اند. ما اینجا برای تشکیل درخت به این شکل عمل می‌کنیم که مقصد می‌شود گروه .

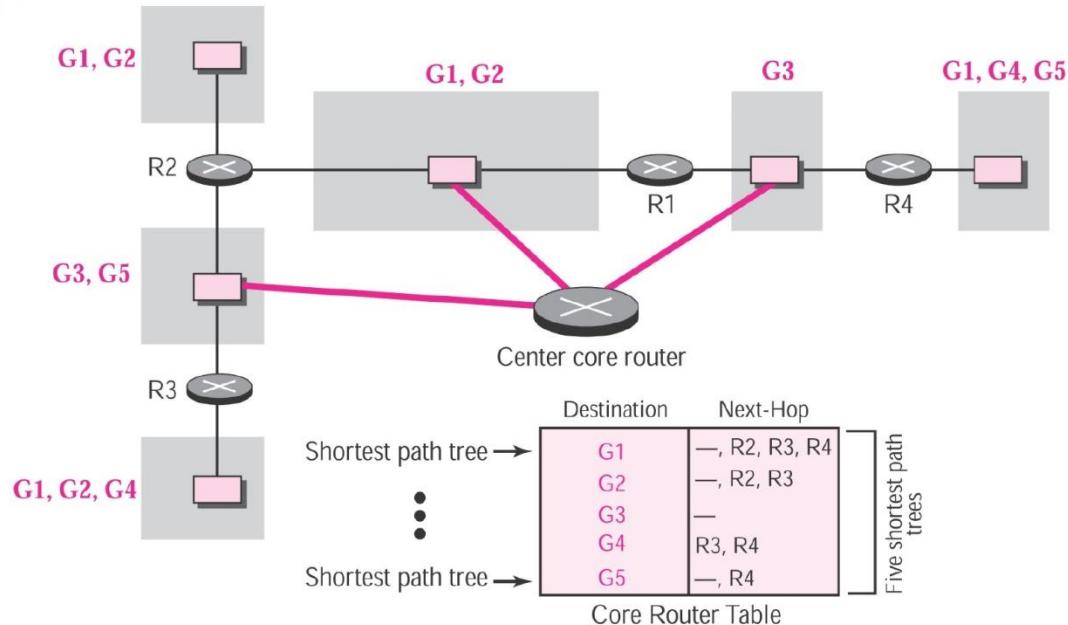
در گروه ۱ (G<sup>۱</sup>) Next Hop کدام هست؟

خط تیره در جدول یعنی اینکه خود روتر وصل هست به نتورک که اعضای گروه یک اونجا قرار دارند و R<sup>۱</sup> و R<sup>۴</sup>

G۲: بلا فاصله از طریق R۲ معنی این است که اگر بسته ای به آدرس مقصد G۵ بخواهد که ارسال شود R۱ باید بسته را هم R۲ ارسال کند و هم به R۴. یعنی دو تا کپی از بسته را Generate می کند و به دو تا روتور R۲ و R۴ ارسال خواهد کرد حالا این درخت پنج تا درخت کوتاهترین مسیر دارد که این درخت ها درخت های مبتنی بر مبدأ هستند.

## چندپوشی در بین چند شبکه

### ۰ درخت مشترک در گروه (group shared)



10 of 37

و اما درخت Group Shared به چه صورت هست؟

اینجا جدول Core Router را داریم.

که یک Core Router Center Network داریم که وصل هست به ۳ تا Core Router Center و وضعیت گروه ها هم به همان شکلی هست که دیدیم.

G<sup>1</sup> از چه راه هایی در دسترس هست در Core Router از طریق ارتباط مستقیم R<sup>2</sup>,R<sup>3</sup>,R<sup>4</sup> و بقیه گروه ها هم داریم که بسته Core Router ها باید کجا ارسال کند در درخت مشترک در گروه هر کدام از این روتراها بسته ای بخواهد به مقصد یک گروهی ارسال کند بسته را به Core Router Center می دهد و این را بر اساس جدول خودش ارسال می کند به روتر ها و در نهایت هم به هاست های موجود در گروه ها می رسد و این درخت وابسته و مبتنی به سورس نیست یعنی نیاز نیست که ما در R<sup>4</sup> یک درخت داشته باشیم در R<sup>2</sup> و R<sup>3</sup> هم یک درخت داشته باشیم اینجا یک درخت داریم و به همه روتراها بسته Multicast را می دهد و اون از طریق درخت سمت مقصد های مختلف پخش می کند.

## چندپخشی در بین چند شبکه

- الگوریتم های چندپخشی : ایجاد مسیرها
- نیازمندی هایی که الگوریتم ها برآورده می کنند:

  - ارسال داده تنها به اعضای گروه
  - بهینه سازی مسیر
  - ایجاد مسیرهای بدون حلقه
  - ارائه توابع سیگنال دهی جهت ایجاد و حفظ گروه
  - عدم ایجاد ترافیک متصرف بر روی بخشی از اتصالات شبکه

بحث مسیرها را در مسیریابی داریم.

نیازمندی هایی که الگوریتم های مسیریابی باید برآورده کند: داده ای که ارسال می شود تنها به اعضای گروه ارسال می شود یعنی به جاهای دیگری (شبکه دیگری) که عضوی از این گروه خاص درش وجود ندارد نباید که ارسال شود.

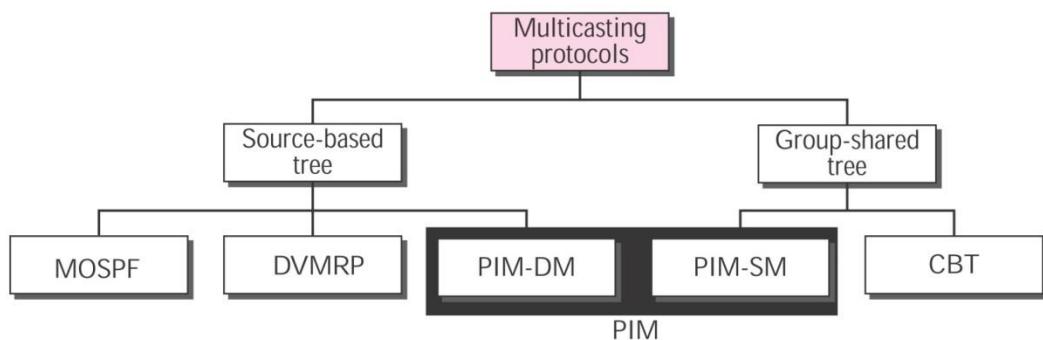
مسیر باید بهینه باشد.

باید بدون حلقه باشد.

در واقع توابع سیگنال دهی باید وجود داشته باشد که برای ایجاد و حفظ گروه از آنها استفاده می کنیم و ترافیک را نباید بر روی بخشی از اتصالات شبکه متمرکز کند این الگوریتم مسیریابی این جوری نباشد که یک لینک خاص یا روتر خاص را بباید به صورت اشتباہ شده ازش استفاده کند در صورتی که قسمت های دیگری شبکه ترافیک کمتری دارند این نیازمندی ها هست که یک الگوریتم مسیریابی چند پخشی باید برآورده کند و این خصوصیات را باید داشته باشد.

## چندپخشی در بین چند شبکه

- پروتکل های مسیریابی چندپخشی
- پروتکل های مختلفی پیشنهاد شده اند، برخی گسترش پروتکل های تک پخشی و برخی دیگر پروتکل های جدید



برای مسیریابی Multicast پروتکل های مختلفی وجود دارد بعضی از این پروتکل ها و پروتکل هایی هستند که مبتنی بر Unicast بوده اند و تغییراتی درش ایجاد شده که تبدیل بشوند به پروتکل Multicast ولی بعضی ها پروتکل های جدیدی هستند که از بیس برای پروتکل های Multicast پیشنهاد شده است بعضی از اینها بر اساس Source Base Tree کار می کنند

دوتا پروتکل Pim هست در واقع Pim دو تا پروتکل داریم که یکی Source Base Tree و دیگری Group Shared هستند.

یک مسیریابی Link State هست و درواقع بهبود گسترش روی مسیریابی Mospf - Unicast - درخت مبتنی بر مبدأ برای ایجاد مسیرها استفاده می کند. اطلاعات گروه ها را در بسته های Link State به همه روتر ها ارسال می کند و در این پروتکل درخت کوتاه ترین مسیر برای هر گروه در هر مسیریاب با استفاده از الگوریتم دایجسترا انجام می شود مشکلی که این الگوریتم به طور عمده دارد این هست که تعداد درخت هایی که ایجاد می شود زیاد هست و مسئله حافظه خواهیم داشت و راه حل این هست که درخته را وقتی بهش احتیاج داریم بسازیم یعنی اطلاعات مربوط به همه نود ها و گروه ها را داریم و این مشکل ایجاد نمی کند ولی این درخت ها هستند که مشکل ایجاد می کنند که می توانیم الگوریتم دایجسترا را زمانی که برای ارسال یک بسته Multicast به یک آدرس نیاز داریم اون موقع درخت را ایجاد کنیم و بعد از استفاده درخت را حذف کنیم.

## چندپخشی در بین چند شبکه

- مسیریابی حالت پیوند چندپخشی
- گسترش مسیریابی حالت پیوند تک پخشی
- استفاده از درخت مبتنی بر مبدأ
- ارسال اطلاعات گروه ها در بسته های حالت پیوند
- ایجاد یک درخت کوتاهترین مسیر برای هر گروه در مسیریاب با استفاده از الگوریتم **dijkstra**
- اشکال : تعداد زیاد درخت ها و مسئله حافظه
- راه حل : ایجاد درخت در زمان نیاز

## جلسه نهم:

### چندپخشی در بین چند شبکه

- مسیریابی حالت پیوند چندپخشی
- گسترش مسیریابی حالت پیوند تک پخشی
- استفاده از درخت مبتنی بر مبدأ
- ارسال اطلاعات گروه ها در بسته های حالت پیوند
- ایجاد یک درخت کوتاهترین مسیر برای هر گروه در مسیریاب با استفاده از الگوریتم **dijkstra**
- اشکال : تعداد زیاد درخت ها و مسئله حافظه
- راه حل : ایجاد درخت در زمان نیاز

گره ها اطلاعات گروه ها را در این الگوریتم Link State Multicast در بسته های Link State ارسال می کند.

در Link State Unicast اطلاعات لینک ها در این بسته ها پیوندی یا ارسال میکنند تا گره های شبکه همه اطلاعات گره های دیگر را داشته باشند. در واقع یک دیدی از کل شبکه و وضعیت لینک های مختلف در شبکه در قسمت های مختلف را داشته باشند و بر اساس آن بتوانند الگوریتم Dijkstra را اجرا کنند.

حالا در Linkstate Multicast هم همین اتفاق می افتد ولی اطلاعات گره ها ارسال می شود و در نهایت یک درخت Shortest Path برای هر گروه در هر مسیریاب تشکیل می شود.

و تشکیل این درخت Shortest Path به وسیله الگوریتم Dijkstra است. درست مثل حالت قبلی.

اشکالی که دارد این است که تعداد زیادی درخت ممکن است ایجاد شود و حافظه زیادی مصرف می شود و ممکن است گرهای حافظه کافی نداشته باشد.

به همین دلیل درخت را در زمان نیاز ایجاد میکنند. هر موقع مسیر نیاز داشت درخت مورد نظر را ایجاد میکند و از آن استفاده میکند و هر موقع کارش تمام شد و پس از مدتی که از آن استفاده نکرد می تواند آن درخت را حذف کند و دوباره اگر نیاز بود الگوریتم را اجرا کند و درخت را ایجاد کند.

## چندپخشی در بین چند شبکه

### MOSPF •

#### • گسترش OSPF

• نیاز به بسته به روز رسانی جدید برای آدرس میزبان های داخل گروه  
**(group membership LSA)**

• ایجاد درختی که شامل همه میزبان های گروه بشود

• پروتکل **data-driven** : دریافت اولین بسته چندپخشی = ایجاد درخت

14 of 37

یک گسترش روی Ospf است . یعنی همان Ospf است که تغییراتی کرده است. به جای اطلاعات لینک ها اطلاعات گروه ها ارسال می شود و یک بسته به روز رسانی جدید برای میزبان Group Membership (Host) هایی که داخل گروه هستند نیاز هست. ( به این میزبان ها میگویند) که این بسته به بسته های قبلی که در Ospf است اضافه می شود.

درختی ایجاد می شود که شامل همه میزبان های گروه بشود.

این پروتکل Data-Driven Multicast است یعنی اولین بسته دریافت بشود درخت ایجاد می شود. درخت از قبل و زمانی که به آن نیازی نیست ایجاد نمی شود و نود ها خود به خود درختی را ایجاد نمی کنند.

## چندپخشی در بین چند شبکه

- بردار فاصله چندپخشی
- پروتکل (RFC 1075) DVMRP
- پروتکل درون AS
- ایجاد درخت per-source, per-group
- تک پخشی را پشتیبانی نمی کند.

Dvmrp(Distance Vector Multicastion Routing Protocol) پروتکلی است با

رiferنس ۱۰۷۵

پروتکل درون As است یعنی درون As اجرا می شود.

درخت میتواند Persource یا Pergroup تشکیل بشود.

این پروتکل تک پخشی (Unicasting) را پشتیبانی نمی کند یعنی با آن نمیتوانیم بسته های تک پخشی را ارسال کنیم.

## چندپخشی در بین چند شبکه

- مسیریابی بردار فاصله چندپخشی
- گسترش مسیریابی بردار فاصله تک پخشی به چندپخشی ساده نیست
- عدم ارسال جدول مسیریابی چندپخشی به همسایه ها
- محاسبه بر اساس جدول مسیریابی تک پخشی
- در واقع جدول مسیریابی تشکیل نمی شود
- استفاده از چهار استراتژی تصمیم گیری

در آنچه هر نod اطلاعات فقط همسایه های خودش را دارد و حالا تعمیم دادنش به گروه کار ساده ای نیست یعنی اطلاعات مورد نیاز شاید به دست نیاید. نکات پیچیده های وجود خواهد داشت.

در Linkstate اطلاعات به همه Node ها می رفت همه Node ها می دانستند که برای هر گروه کدام Host ها عضو گروه هستند و Host ها از چه مسیری در دسترس هستند و همه اطلاعات شبکه را داشتند ولی در Dvmrp فقط اطلاعات Node چون همسایه خودش را دارد تعمیم این مسیر به چند پخشی ساده نیست.

جدول مسیریابی چندپخشی هم به همسایه ها ارسال نمی شود و محاسبات بر اساس جدول مسیریابی تک پخشی است.

اگر ما Node های گروه ها را بدانیم و بر اساس مسیریابی تک پخشی به هر کدام از این Node ها محاسبات انجام دهیم میتوانیم مسیرهای دسترسی به Node های عو گروه را به دست بیاوریم.

در این مسیریابی بردار فاصله چند پخشی، جدول مسیریابی برای یک گروه تشکیل نمی شود. و در این پروتکل از ۴ استراتژی تصمیم گیری استفاده می کنیم.

## چندپخشی در بین چند شبکه

- مسیریابی بردار فاصله چندپخشی
- استراتژی غرق کردن (flooding)
- در واقع یک روش همه پخشی است
- کارایی خوبی ندارد
- ممکن است حلقه ایجاد شود

### (a) استراتژی غرق کردن (Flooding)

یک استراتژی همه پخشی است یعنی ما بسته ها را به صورت Multicast به مقصد تعدادی گروه نمیفرستیم.

بسته را به همه گروه‌ها می‌فرستیم، آنها یکی که عضو گروه هستند بسته را بر میدارند و آنها یکی که عضو گروه نیستند بسته را حذف می‌کنند.

این روش کارایی خوبی ندارد، که یکی از اشکالاتش است چون ممکن است بسته به مسیرهای ارسال بشود که نیاز نیست ارسال شود و ممکن است حلقه ایجاد بشود یعنی ممکن است بسته‌ای که به گره‌ای می‌رسد از یک گره دیگرهم دوباره بررسد.

اگر از Sequense Number استفاده نشود ممکن است بسته تکراری ارسال شود و بسته تکراری تشخیص داده نشود

(۲) استراتژی Rpf(Reverse Path Forwarding) یا جلورانی مسیر معکوس یکی از بسته‌ها را Forward کنیم یعنی بسته‌ای که از مسیر بهینه از مبدا رسیده است یعنی نوع مسیر بهینه خودش به مبدا بسته را میداند و اگر از مسیری غیر از آن بسته را دریافت کرد بسته را Forward نمی‌کند. فقط بسته‌ای که از پورت Shortest Path آمد را دریافت کند.

تشخیص تکرار را به این صورت انجام میدهیم نه با Sequence Num یعنی تشخیص بر اساس پورت ورودی یعنی بسته از پورت مسیر بهینه بسته اصلی است و بقیه تکراری‌اند.

## چندپخشی در بین چند شبکه

- مسیریابی بردار فاصله چندپخشی

### استراتژی جلورانی مسیر معکوس (Forwarding (RPF))

- جلورانی فقط یکی از بسته ها : بسته ای که از مسیر بهینه از مبدأ رسیده است

- تشخیص با استفاده از جدول مسیریابی تک پخشی

- پیشگیری از حلقه

همان Broadcast است که یک درجه بهبود داده است و آن هم این است که بسته های تکراری را به این شکل تشخیص میدهد و آن ها را Forward نمیکند.

تشخیص مسیر بهینه هم با استفاده از جدول مسیریابی تکپخشی است این کار باعث جلوگیری از حلقه می شود.

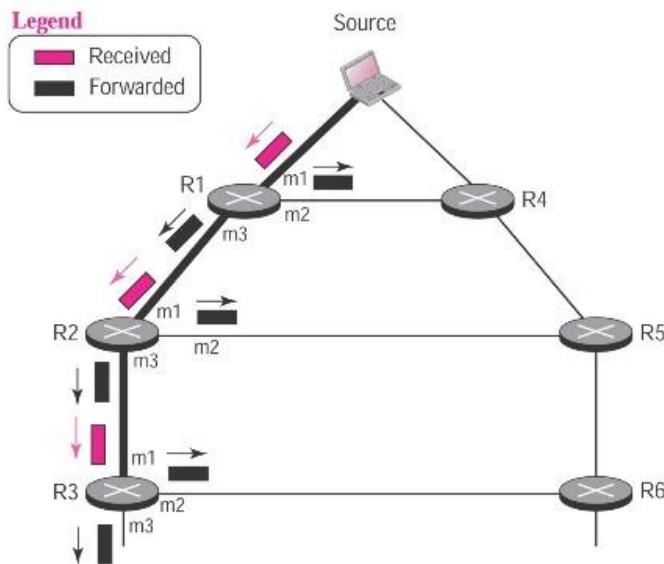
با این روش محاسبات اضافه نمیخواهد چیزی به عنوان Sequence Num به بسته اضافه کنیم که دیتای اضافی داخل شبکه وارد کنیم.

فقط با مشاهده اینکه بسته از کدام مسیر رسیده است می شود تشخیص داد بسته جدید است یا تکراری است.

## چندپوشی در بین چند شبکه

### • مسیریابی بردار فاصله چندپوشی

### ۰ استراتژی جلوگاهی مسیر معکوس (Forwarding (RPF))



19 of 37

مثلاً روتر  $R_1$  مسیر بهینه با  $Source$  پورت بالایی است بسته ای که از این پورت دریافت میکند  $Forward$  را میکند.

اگر از  $R_4$  هم یک بسته دریافت کند دیگر آن را  $Forward$  نمیکند چون از پورت مسیر بهینه  $Flooding$  بسته تکراری دریافت می شود و حلقه ایجاد می شود.

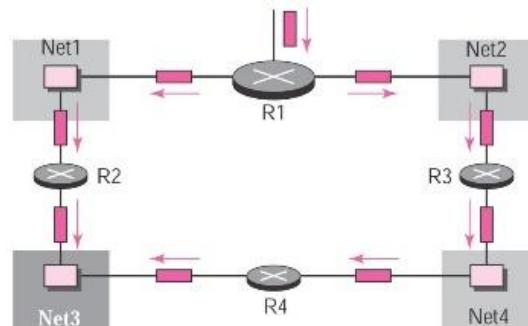
## چندپخشی در بین چند شبکه

• مسیریابی بردار فاصله چندپخشی

• استراتژی همه پخشی مسیر معکوس (Broadcasting (RPB))

• تضمین دریافت حداقل یک کپی از بسته بدون ایجاد حلقه توسط RPF

• عدم تضمین فقط یک کپی توسط RPF



20 of 37

۳) استراتژی (Rpb)(Reverse Path Boarding)

در Rpf ما میتوانیم تضمین کنیم حداقل یک کپی از بسته بدون ایجاد حلقه توسط همه نود های داخل شبکه دریافت خواهد شد. ولی تضمین نمیکند که فقط یک کپی دریافت شود ممکن است چندین کپی دریافت شود.

مثال دریافت بسته تکراری

بسته ای که به  $R_1$  آمده واز  $R_1$  به  $Net_2$  و  $Net_3$  ارسال شده یعنی  $R_2$  و  $R_3$

$R^3$  به  $R^4$  ارسال کرده است.

$R^2$  و  $R^4$  هر دو به Net<sup>۳</sup> ارسال کرده اند. این اشکال Rpf است که بسته تکراری را می‌گیرد.

## چندپخشی در بین چند شبکه

- مسیریابی بردار فاصله چندپخشی

### استراتژی همه پخشی مسیر معکوس (Broadcasting (RPB))

- دلیل بسته تکراری در RPF: عدم تشکیل درخت

- در RPB: هر مسیریاب برای بسته‌های هر مبدأ برای برخی از شبکه‌ها حکم والد (parent) را دارد

- بسته‌ها را فقط به شبکه‌هایی که والد آنهاست ارسال می‌کند

دلیل ایجاد بسته تکراری این است که درختی تشکیل نمی‌شود فقط ما Forwarding انجام می‌

دهیم.

در Rpb هر مسیریاب برای بسته‌های هر مبدأ برای برخی از شبکه‌ها والد (Parent) محسوب

می‌شود. روتربسته‌ها را فقط به شبکه‌هایی می‌دهد که والد آنها است.

در مثال Net<sup>۳</sup> اگر یکی از  $R^2, R^4$  به عنوان والد خودش در نظر بگیرد آن دیگری که والد

Net<sup>۳</sup> نیست دیگر به Net<sup>۳</sup> بسته ارسال نمی‌کند.

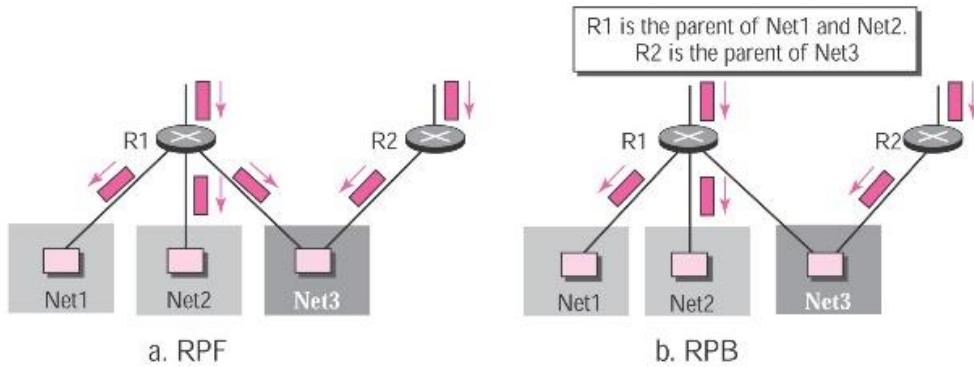
یک روتر ممکن است برای بسته های مبدا خاصی حکم والد را داشته باشد و برای بسته با مبدا های دیگر نداشته باشد.

یک روتر می تواند برای یک بسته، والد یک Node باشد و برای بسته دیگر، والد Node دیگر باشد.

## چندپخشی در بین چند شبکه

### • مسیریابی بردار فاصله چندپخشی

### • استراتژی همه پخشی مسیر معکوس (Broadcasting (RPB))



در  $R_1, R_2, R_{fp}$  بسته  $Net^3$  به  $R_1, R_2, R_{fp}$  بسته خواهد داد و بسته تکراری خواهد شد.

در  $R_1, R_2, R_{fb}$  والد  $Net^1$  و  $Net^2$  و  $Net^3$  است و والد  $Net^3$  بسته  $Net^3$  را از آن دریافت میکند. یعنی  $R_1$  بسته ای به  $Net^3$  نخواهد داد.

## چندپخشی در بین چند شبکه

- مسیریابی بردار فاصله چندپخشی

### استراتژی همه پخشی مسیر معکوس (Broadcasting (RPB))

- نحوه تعیین والد:

- مسیریابی که کوتاهترین مسیر تا مبدأ را دارد
- در صورت وجود بیش از یک مسیریاب: انتخاب مسیریابی با آدرس کوچکتر

برای تضمین والد ها از الگوریتم مسیریابی کوتاه ترین مسیر (Distance Vector) استفاده می شود.

اگر از یک Net به یک Source بیش از یک مسیر بهینه وجود داشته باشد انتخاب مسیریابی با آدرس کوچکتر انجام می شود یعنی روتر ای به عنوان والد انتخاب می شود که عدد آدرس کوچکتری داشته باشد.

## چندپخشی در بین چند شبکه

• مسیریابی بردار فاصله چندپخشی

• استراتژی چندپخشی مسیر معکوس (Reverse Path Multicasting (RPM))

• RPB همه پخش می کند، کارا نیست

• برای چندپخشی دو رویه استفاده می شود: هرس کردن (pruning) و پیوند زدن (grafting)

در واقع در Rpb هم ما Broadcasting انجام می دهیم یعنی Select Net ها را نمیکنیم.

یعنی نمیگوییم Node های متعلق به این گروه توی این Net هستند پس به این Net بسته را بدهیم و به اونی که نیست بسته را ندهیم.

این Broadcasting کارا نیست چون بخشی از شبکه بسته هایی را دریافت می کند که نیاز نیست آنجا برود و شبکه شلوغ می شود.

۴) استراتژی Rpm(Reverse Path Multicasting)

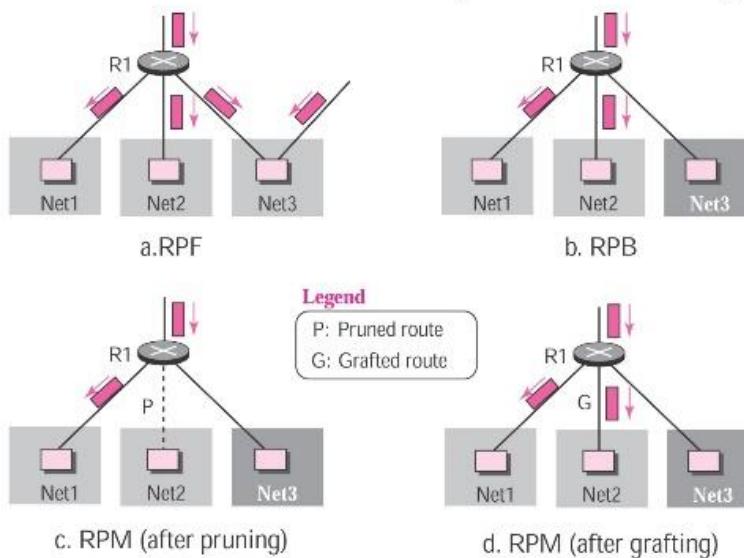
این استراتژی از دو رویه استفاده می کند. ۱) هرس کردن (Pruning) ۲) پیوند زدن (Grafting)

با این دو کار به جای Broadcasting میتوانیم Multicasting انجام دهیم . یعنی بسته را فقط به Net هایی بدهیم که در آن ها Node هایی هستند که عضو گروه هستند.

## چندپخشی در بین چند شبکه

### • مسیریابی بردار فاصله چندپخشی

#### • استراتژی چندپخشی مسیر معکوس (Multicasting (RPM))



25 of 37

فرض کنید بعد از اینکه Net<sup>۲</sup> را خودش را شناخت Net<sup>۱</sup> ای مثل Node هیچ Net<sup>۲</sup> ای عضو گروه نداشته باشد پس برای آن عمل هرس کردن باید اتفاق بیفتد. یعنی روتر R<sup>۱</sup> اتصال اش را با Net<sup>۲</sup> قطع کند.

به این عمل Pruning و به آن روت Prun Router می گوییم.

در شکل Rpf<sup>۳</sup> از دو طرف بسته می گیرد.

در Rpb<sup>۳</sup> از والدش بسته می گیرد.

در  $Net^3$  ،  $Rpm$  گره ای داخل گروه ندارد پس از  $R1$  حذف می شود. ممکن است بعداً یکی از  $Net$  های داخل گروه شود پس عمل پیوند زدن صورت می گیرد. یعنی دوباره خودش را به  $R1$  پیوند می زند و بسته دریافت می کند.

## چندپخشی در بین چند شبکه

### • مسیریابی بردار فاصله چندپخشی

### • استراتژی چندپخشی مسیر معکوس (Multicasting (RPM))

• هرس کردن : حذف مسیرهایی که میزبان در این گروه ندارند

• با ارسال بسته های **prune** از طرف شبکه به مسیریاب

• دریافت بسته **prune** از همه مسیرهای رو به پایین ← ارسال بسته **prune** به مسیریاب رو به بالا

هرس کردن یعنی حذف مسیرهایی که میزبان در این گروه ندارند مثل مثال بالا.

هرس کردن با ارسال بسته **Prune** از طرف شبکه به روترا اتفاق می افتد.

اگر از یک روترا از همه مسیرهای رو به پایین اش بسته **Prune** دریافت کرد باید یک بسته

به روترا بالایی خودش بفرستد. یعنی میگوید من عضوی از گروه را در زیرمجموعه ام ندارم.

## چندپخشی در بین چند شبکه

• مسیریابی بردار فاصله چندپخشی

• استراتژی چندپخشی مسیر معکوس (Multicasting (RPM))

• پیوند زدن : ارسال بسته graft به مسیریاب رو به بالا

• دسترسی بر RIP مبتنی بر DVMRP .

پیوند زدن هم با ارسال بسته Graft به روتر بالایی انجام می شود.

وقتی یک بسته Graft از شبکه پایینی به مسیریاب بالایی برسد در صورتی که روتر قبل Prune شده باشد مسیر بهش اضافه می شود.

• دسترسی بر Rip مبتنی بر Dvmrp .

این پروتکل ها گسترش یافته پروتکل های قبلی هستند.

## چندپخشی در بین چند شبکه

- مسیریابی بردار فاصله چندپخشی
- پروتکل درخت مبتنی بر هسته (Core-Based Tree (CBT))
- استفاده از یک مسیریاب مرکزی به عنوان ریشه درخت
- تقسیم AS به یک سری محدوده (region) و یک هسته برای هر محدوده

Multicasting Cbt(Core Base Tree) پروتکل درخت مبتنی بر هسته است و برای تعريف شده است.

در این پروتکل یک روتر مرکزی به عنوان Root در درخت انتخاب می شود و از آن برای استفاده می شود. Multicasting

در این پروتکل Region را به یک سری Region تقسیم میکنیم و یک هسته برای هر Region نظر می گیریم. هسته همان روترا اصلی است.

## چندپخشی در بین چند شبکه

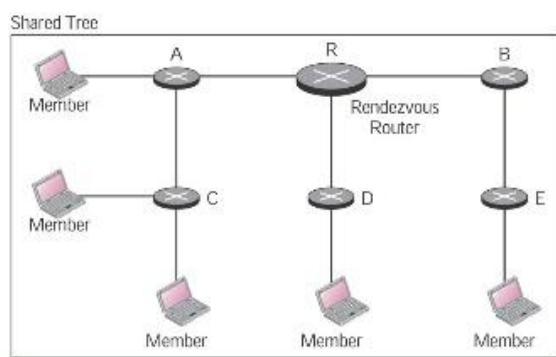
### • مسیریابی بردار فاصله چندپخشی – CBT

#### ◦ شکل گیری درخت:

◦ ارسال بسته الحاق (join) به گروه بعد از مشخص شدن مسیریاب مرکزی

◦ ذخیره اطلاعات بسته های الحاق توسط مسیریاب های میانی

◦ ترک گروه: ارسال بسته ترک گروه به مسیریاب مرکزی



29 of 37

بعد از اینکه روتر اصلی مشخص شد هر Join Node بسته الحاق یا Join به روتر مرکزی بدهد.

اطلاعات بسته های الحاق توسط روتر های میانی هم ذخیره می شود.

در شکل روتر R روتر مرکزی است و هر کدام از Host ها می خواهد عضو گروه شود بسته Join را می فرستد.

اگر Member وسطی بسته ای را به R بفرستد اطلاعاتش را در خودش ذخیره میکند.

برای ترک گروه هم بسته ترک گروه را به روتر میفرستد و روترهای میانی اطلاعات مربوط به ترک کردن گروه توسط آن Node را ذخیره می کنند.

مثلا اگر Member وسطی بخواهد گروه را ترک کند دیگر روتر D بسته های این گروه را Forward نخواهد کرد.

## چندپخشی در بین چند شبکه

• مسیریابی بردار فاصله چندپخشی – CBT

• دو تفاوت بین CBT و MOSPF و DVMRP با

۱- در دو پروتکل اول درخت از ریشه شکل می گیرد اما در CBT از برگ

۲- در DVMRP درخت شکل گرفته و هرس می شود ولی در CBT درخت با پیوند تشکیل می شود

تفاوت CBT با Mospf و Dvmrp :

در دوپروتکل اول درخت از ریشه شکل می گیرد اما در Cbt درخت از برگ شکل می گیرد و هر کدام از Node ها که درخواست Join می دهند روترهای داخل مسیر این اطلاعات را ذخیره می کنند و درخت را تشکیل می دهند.

در Dvmrp درخت اول ایجاد می شود و بعد عمل هرس کردن انجام می شود یعنی مسیرهایی که عضوی در گروه ندارند درخواست هرس شدن می دهند ولی در Cbt درخت با ایجاد پیوند

تشکیل می شود یعنی از اول درختی وجود ندارد و هر Node که درخواست Join کردن به گروه را میدهد Link ها به ترتیب وارد درخت می شود و پیوند میخورند.

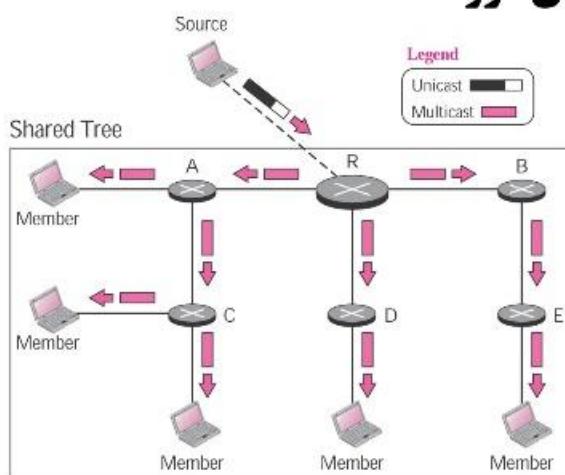
## چندپخشی در بین چند شبکه

### • مسیریابی بردار فاصله چندپخشی – CBT

- ارسال بسته های چندپخشی:

- ارسال بسته به مسیریاب مرکزی

- پخش بسته به مسیریاب های گروه



31 of 37

زمانی که در Cbt یک بسته میخواهد Multicast ارسال بشود بسته باید به روتر مرکزی (یعنی روتری که به عنوان روتر Root است) ارسال شود و بعد روتر مرکزی مسیربه همه Member های گروه را دارد و بسته آنجاها ارسال می شوند.

اگر مثلا Member وسطی از گروه خارج بشود دیگر روتر به آن سمت بسته ای ارسال نمی کند.

## چندپخشی در بین چند شبکه

• چندپخشی مستقل از پروتکل (PIM)

• دو پروتکل مستقل : **PIM- PIM-DM(Dense Mode) و DM(Sparse Mode)**

• هر دو گسترش یافته پروتکل تک پخشی

روش بعدی Pim(Protocol Independent Multicast) است که یک روش مستقل از پروتکل است.

دراین روش دو پروتکل داریم :

Pim-Dm(Dense Mode) (۱)

Pim-Sm(Sparse Mode) (۲)

هر دو گسترش یافته پروتکل Unicast هستند.

## چندپخشی در بین چند شبکه

### :PIM-DM •

• مناسب برای زمانی که اکثر مسیریاب ها در چندپخشی شرکت می کنند  
**(حالت چگال)**

• پروتکل همه پخشی مناسب تر است

• **PIM-DM** : مبتنی بر درخت مبتنی بر مبدأ

• استفاده از **RPF** و هرس / پیوند، مشابه **DVMRP**

• عدم وابستگی به یک پروتکل تک پخشی خاص

Multicasting یا حالت چگال مناسب زمانی است که بیشتر روترهای شبکه در Pim-Dm شرکت می کنند. در چنین حالتی اگر Broadcasting انجام دهیم شاید بهتر باشد.

Pim-Dm مبتنی بر درخت مبتنی بر مبدأ است.

از Rpf برای Forwarding استفاده می کند و هرس و پیوند را مثل Dvmrp انجام می دهد و به پروتکل تک پخشی خاصی وابسته نیست.

## چندپخشی در بین چند شبکه

### :PIM-SM •

- مناسب برای زمانی که احتمال کمی وجود دارد که هر مسیر یاب چندپخشی شرکت می کنند (حالت پراکنده)
- پروتکل همه پخشی مناسب نیست، پروتکلی مثل CBT مناسب تر است

### PIM-DM •

- مشابه CBT، اما عدم نیاز به تصدیق بسته های الحاق
- ایجاد نقاط مرکزی (RP(Rendezvous Point)) پشتیبان
- قابلیت تغییر به درخت مبتنی بر مبدأ برای مناطق چگال

34 of 37

Pim-Sm مناسب برای زمانی است که تعداد روترهای کمی در Multicast شرکت میکنند(حالت پراکنده).

در چنین حالتی Broadcasting مناسب نیست چون بسته به جاهایی از شبکه می رود که نباید برود و Overhead زیادی میشود و عدم کارایی بالا می رود.

در این حالت پروتکل Cbt بهتر است.

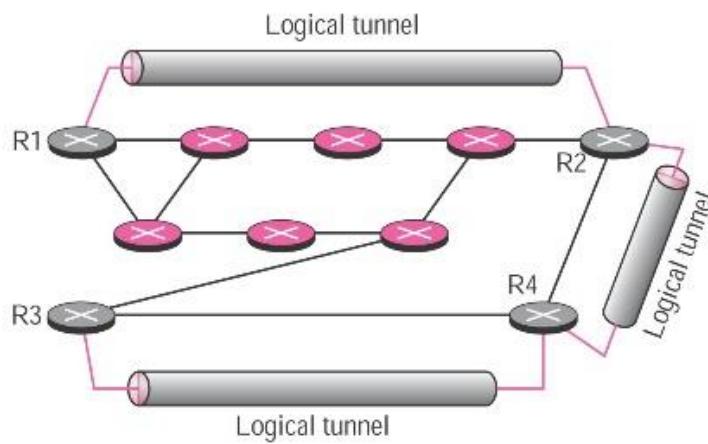
Pim-Sm مبتنی بر درخت مشترک در گروه است. مثل Cbt است ولی به تصدیق (Ack) بسته های الحاقی نیازی ندارد. مثل Cbt نقاط مرکزی (Rp(Rendezvous Point) به وجود می آید.

برای مناطق متراکم (چگال) این امکان وجود دارد Pim-Sm به Pim-Dm تغییر کند و یک Multicast Broadcast عمل کند بقیه به صورت بخشی از شبکه به صورت

## چندپخشی در بین چند شبکه

### : (Multicast Backbone) MBONE •

- افزایش نیاز به چندپخشی بدلیل ارتباطات چندرسانه ای و بلاذرنگ
- عدم گسترش مناسب مسیریاب های چندپخشی
- یک راه حل : تونل زنی



35 of 37

### :Mbone(Multicast Backbone)

نکته ای که وجود دارد این است که نیاز به چندپخشی به دلیل ارتباطات چندرسانه ای و بلاذرنگ افزایش پیدا کرده است یعنی خیلی جاها نیاز است که Multicast انجام شود.

مثال های Multicast روز به روز پر کاربرد تر می شود ولی روتراها با قابلیت Multicast خیلی

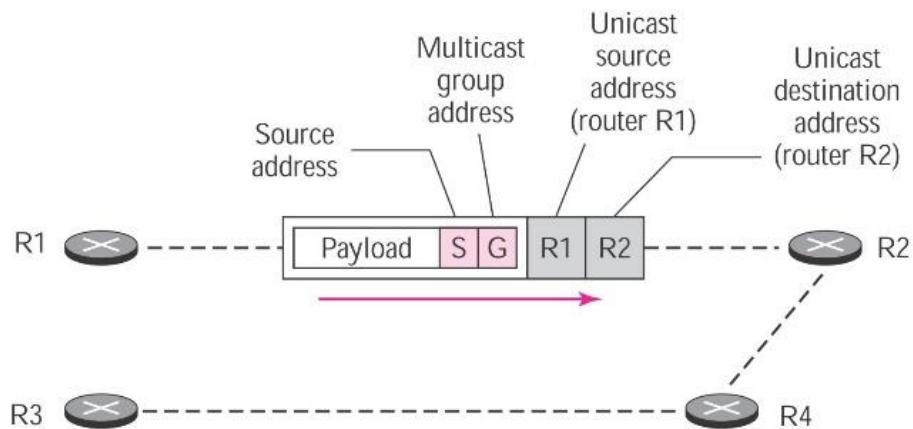
گسترش پیدا نکرده اند یعنی به درستی و متناسب گسترش پیدا نکرده اند.

یک راه حل برای این مشکل که یک راه موقتی است استفاده از تونل زنی (Tunelling) است.

روترهایی که Multicast Enable هستند را با این روش می توانیم دور بزنیم.

## چندپخشی در بین چند شبکه

### : (Multicast Backbone) MBONE •



36 of 37

در واقع بسته Multicast وقتی می خواهد از یک بخشی از شبکه که روتراهای Multicast در آنجا وجود ندارند عبور کند می توانیم داخل یک بسته Unicast Encapsulate کنیم و آن را به روتر بفرستیم.

در شکل می بینید که بسته های Unicast می توانند از روترهای Multicast هم عبور کنند. به این روش Tunelling میگویند.

## پروتکل های مسیریابی

### مسیریابی چندپخشی (Multicast) پروتکل IGMP

پروتکل مدیریت گروه در اینترنت

مالتی کست : ارسال پیام به چند هاست.

یونی کست: ارسال یک پیام از مبدا به یک مقصد مشخص

براد کست : ارسال پیام از یک مبدا به همه هاست ها

مالتی کستینگ با یونی کست ممکن است . در مالتی کست یک کپی ارسال میشود و مسیر یاب ها باید این پیام را روی پورت هایی ارسال کنند که آن هاست در گروه هست .

- چندپخشی : ارسال یک پیام به چند میزبان
- ارسال یک کپی توسط فرستنده
- مسیریاب های چندپخشی باید لیست گروه ها را بدانند
- مسیریاب های چندپخشی باید اطلاعات چندپخشی را جمع کرده و به سایر مسیریاب های چندپخشی ارسال کنند
- دو سطح در جمع آوری اطلاعات:
  - محلی
  - عمومی

و این اطلاعات چند پخشی را مسیر یاب ها باید داشته باشند و به همسایگانش هم بدهد تا بتواند به پور های مد نظر هدایت کنند.

## IGMP

- جمع آوری اطلاعات به صورت محلی توسط مسیریاب متصل به شبکه :  
**IGMP پروتکل**
- پخش عمومی اطلاعات جمع آوری شده : توسط پروتکل های مسیریابی  
چندپخشی
- وظیفه IGMP اصلاح و نمایش اطلاعات اعضای گروه است



5 of 33

وظیفه Igmp مدیرت عضویت اعضا در گروه در روتر ها است. که جایگاه آن در تصویر مشخص است . در واقع از بسته های Igmp برای Ip استفاده میشود.

- مدیریت گروه
- مدیریت عضویت در گروه
- یک یا چند مسیریاب در هر شبکه برای توزیع بسته های چندپخشی
- IGMP : ارائه اطلاعات عضویت به مسیریاب های چندپخشی
- IGMP : ایجاد لیست گروه ها برای هر مسیریاب چندپخشی
- IGMP سه نسخه دارد
- نسخه ۱ و ۲ : Any Source Multicast (ASM)
- نسخه ۳ : Source Specific Multicast (SSM) ، گیرنده می تواند لیستی از گره های مبدا را برای دریافت بسته چندپخشی انتخاب کند

6 of 33

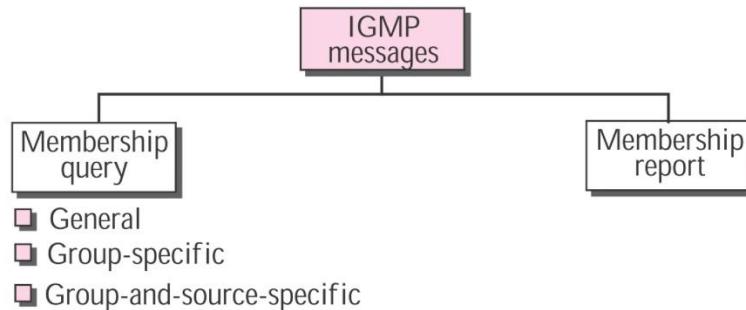
در نسخه ۳ مبدا بسته اهمیت دارد.

در نسخه ۲ قدرت انتخاب نداریم برای اینکه مثلا بسته از فلان مبدا ها را دریافت نکنیم.

**۰ پیام های IGMP**

**دو نوع :** پیام سوال عضویت (membership query) و پیام گزارش عضویت (membership report)

**پیام سوال عضویت :** عمومی، وابسته به گروه، وابسته به گروه و مبدأ



7 of 33

**۰ قالب پیام سوال عضویت**

**ارسال شده توسط مسیریاب جهت یافتن اعضای گروه در شبکه**

0	8	16	31
Type: 0x11	Response code	Checksum	
Group address			
Resv	S   QRV	QQIC	Number of sources (M)
Source Address (1)			
Source Address (2)			
⋮			
Source Address (N)			

مسیر یاب این پیام ها را ارسال میکند تا بتوند هاست های عضو گروه را پیدا کند. هاست ها با  
ممبر شیپ ریپورت مسیج می گوبد که من عضو گروه هستم . اگر نفرستد هم یعنی عضو گروه  
نیست.

یک بخش تایپ دارد که هگز است و راجع به آنها صحبت میکنیم.

## IGMP

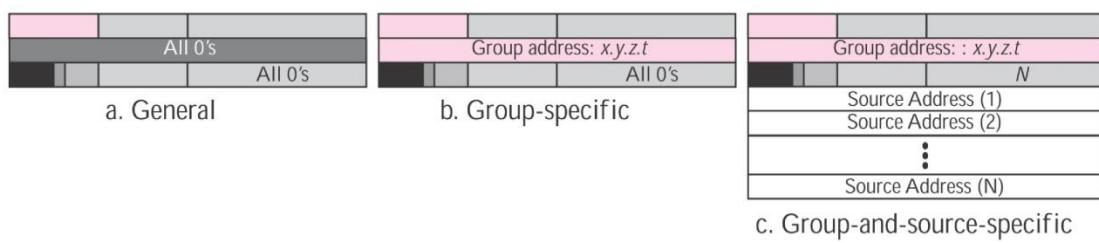
- **Type :** نوع پیام
- **Maximum Response Code :** تعریف زمان پاسخ
- **Checksum :** کد تشخیص خطا
- **Group Address :** آدرس گروه
- **suppress flag :S** : عدم به روز رسانی تایمر
- **QRV :** بررسی استحکام
- **QQIC :** بازه های پرس و جو
- **Number of sources (N) :** تعداد آدرس های مبدا در پیام
- **Source Addresses :** آدرس های مبدا

9 of 33

چقدر باید صبر کنیم تا به هاست جواب بدیم Max Response Code

در اگر S یک باشد تایمر آپدیت نمی شود.

## • سه شکل پیام های پرس و جو



در کوئری جنرال همه‌ی گروپ ادرس آن صفر است . سورس ادرس آن نیز صفر راست

گروپ اسپیسیفیک چون فقط به گروه وابسته است نه به سورس گروه آدرس دارد و تعداد سورس‌ها صفر است.

در Group–Source Specific کوئری آدرس را داریم و سورس آدرس هم به تعدادی است که به آن تعداد سورس آدرس داریم.

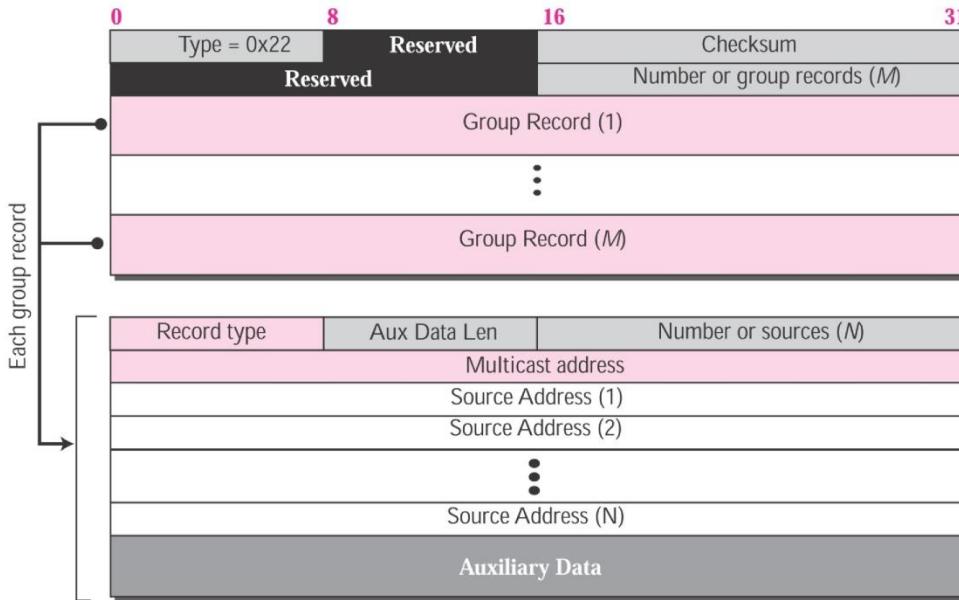
برای انواع کوئری‌ها قالب اسلاید قبل قابل استفاده است.

در اولی هر هاستی باید پیام دهد که عضو کدام گروه است.

در دومی روتر دارد میپرسد کی ها عضو این گروه اند.

در سومی روتر میپرسد کی ها عضو این گروه اند و از این سورس‌ها پیام دریافت میکنند.

## ۰ قالب پیام گزارش عضویت



11 of 33

در این اسلاید فرمت پیام ریپورت رو داریم. در آن تعداد گروپ رکورد ها مشخص است . چون یک هاست ممکن است عضو چند گروه باشد .

طول دیتای کمک بعد از سورس می آید و سپس تعداد سور آدرس و خود سورس آدرس و در آخر هم دیتای کمکی می آید.

در Igmp روتراز هاست ها میپرسد کی ها عضو این گروه اند و از این سورس ها پیام دریافت میکنند و هاست ها اعلام میکنند و روتراز جدول خودش را کامل میکند تا بتواند بسته را به پورت مناسب برای دریافت در مقصد هدایت کند

## IGMP

### • پروتکل IGMP در میزبان ها

#### • وضعیت سوکت

- شروع مدیریت گروه ها در پروسه متصل به یک واسطه
- هر پروسه رکوردی به ازای هر گروه که پیام هایش را می خواهد دریافت کند دارد

#### • رکورد در یکی از دو حالت : شمول (include)، امتناع (exclude)

- شمول : لیست مبدأهایی که بسته آنها را دریافت می کند
- امتناع : لیست مبدأهایی که بسته آنها را دریافت نمی کند

12 of 33

هر پروسس می تواند به چند کارت شبکه متصل باشد .

**کامنت :** بچه ها این اسلاید و مبحث به نظرم خیلی مهمه خوب بخونید (نویسنده ای این

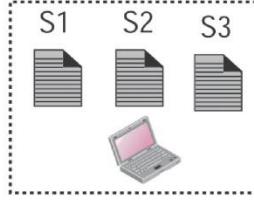
**کامنت علی خیر**

## • مثال:

States Table

**Legend** S: Socket  
a, b, ...: Source addresses

Socket	Multicast group	Filter	Source addresses
S1	226.14.5.2	Include	a, b, d, e
S2	226.14.5.2	Exclude	a, b, c
S2	228.24.21.4	Include	b, c, f
S3	226.14.5.2	Exclude	b, c, g
S3	228.24.21.4	Include	d, e, f



13 of 33

سه سوکت داریم و یکسری سورس از A تا G.

S1 یک رکورد داد Include هست و سورس ها هم A-B-D-E هستند این یعنی سوکت

S1 بسته هایی از گروه ۲۲۶.۱۴.۵.۲ از سورس های A-B-D-E در یافت کند. و از سایر

سورس ها نمیخواهد دریافت کند. با سایر مالتی کست گروه ها هم کاری ندارد.

S2 دو رکورد دارد یک Include یک Exclude که در Include می خواهد فقط بسته

های سورس B-C-F را از مالتی کست گروپ ۲۲۸،۲۴،۲۱،۴ را در یافت کند و در Exclude

فقط بسته های سورس A-B-C را نم خواهد دریافت نکند و سایر بسته ها را می خواهد دریافت کند ۱۴.۵.۲.

در S<sup>۳</sup> هم مانند ...S<sup>۲</sup>.

## IGMP

### • وضعیت واسط

- امکان اشتراک یک گروه چندپخشی بین دو یا چند سوکت
- لازم است که واسط هم وضعت را نگه دارد
- در ابتدا خالی است و در طول زمان تغییر می کند
- فقط یک رکورد برای هر گروه چندپخشی
- در مثال قبل : فقط دو رکورد
- مشکل در ترکیب رکوردهای : لیست مبدأها

هر سوکت رکوردی به ازای هر گروهی که میخواهد پیام را از آنها در یافت کند دارد و آنها را از پروسس ها میاورد.

امکان اشتراک یک گروه مالتی کست بین یک یا دو گروه وجود دارد.

اینترفیس باید وضعیت داشته باشد . که در ابتدا خالی است.

در مثال قبل اینترفیس ما فقط دو رکورد میتواند داشته باشد.

ترکیب سورس آدرس ها یکم سخته فقط.

## IGMP

- ۰ اگر یک گروه چندپخشی چند لیست رکورد متفاوت داشته باشد
- ۰ اگر یکی از رکوردها حالت امتناع داشته باشد رکورد حاصل حالت امتناع دارد
  - ۰ محاسبه اشتراک لیست امتناع
  - ۰ محاسبه اختلاف آن با لیست شمول
- ۰ اگر همه رکوردها حالت شمول داشته باشند رکورد حاصل حالت شمول دارد و  
اجتماع همه رکوردها است

برای تبدیل ۲ رکورد به هم باید به نحوه‌ی که در اسلاید بالا گفته شده باید عمل کرد:

ترجمه: اگر یک گروه چند لیست رکورد متفاوت داشته باشد هم Include هم Exclude .. اگر

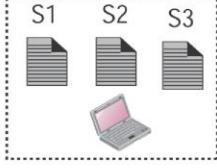
حداقل یک اکس کلود داشتیم جواب نهایی اکسکلود است . در این حالت:

اشتراک لیست Exclude ها را میگیریم و اختلافشان را با لیست Include ها حساب میکنیم تا به جواب برسیم.

اگر همه‌ی رکورد ها Include باشند: رکورد حاصل Include است و لیست Include ها را اجتماع میکنیم (همه‌ی موجودی Include ها)

مثال اسلاید قبل ۲۲۸.۲۴.۲۱.۴ دو رکورد Include داریم آنها را اجتماع میکنیم

در مثال بالا برای S2 اشتراک لیست Exclude ها را محاسبه میکنیم که B و C مشترک است . حالا اختلاف Exclude ها با Exclude ها را محاسبه میکنیم چون B در لیست **Exclude C** ها هست اختلاف فقط C میشود و جواب نهایی ما **Include** است.



**IGMP**

**States Table**

**Legend** S: Socket  
a, b, ...: Source addresses

Socket	Multicast group	Filter	Source addresses
S1	226.14.5.2	Include	a, b, d, e
S2	226.14.5.2	Exclude	a, b, c
S2	228.24.21.4	Include	b, c, f
S3	226.14.5.2	Exclude	b, c, g
S3	228.24.21.4	Include	d, e, f

**• مثال :**

**: 226.14.5.2**

**exclude source list = {a, b, c} . {b, c, g} - {a, b, d, e} = {c}**

**: 228.24.21.4**

**include source list = {b, c, f} + {d, e, f} = {b, c, d, e, f}**

## جلسه دهم:

The diagram shows a legend where S: Socket and a, b, ...: Source addresses. It includes a States Table with columns: Socket, Multicast group, Filter, and Source addresses. The table has five rows:

Socket	Multicast group	Filter	Source addresses
S1	226.14.5.2	Include	a, b, d, e
S2	226.14.5.2	Exclude	a, b, c
S2	228.24.21.4	Include	b, c, f
S3	226.14.5.2	Exclude	b, c, g
S3	228.24.21.4	Include	d, e, f

• مثال :

- برای گروه 226.14.5.2:  
**exclude source list = {a, b, c} . {b, c, g} - {a, b, d, e} = {c}**
- برای گروه 228.24.21.4:  
**include source list = {b, c, f} + {d, e, f} = {b, c, d, e, f}**

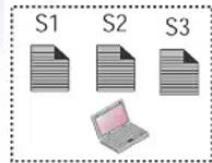
16 of 33

بریم ببینیم اسلاید بعدی چیه؟ جواب همین مثال هست. ببینید جدول State ها برای سوکت ها رو میبینید که لیست اول برای ۲۲۶.۱۴.۵.۲ یک Exclude Link هست Interface و لیست دوم برای ۲۲۸،۲۲۴،۲۱،۴ یک Include Link C هست. و Source Address F هست. اون B,C,D,E,F هاست.

## IGMP

• مثال :

States Table



Legend

S: Socket  
a, b, ...: Source addresses

Socket	Multicast group	Filter	Source addresses
S1	226.14.5.2	Include	a, b, d, e
S2	226.14.5.2	Exclude	a, b, c
S2	228.24.21.4	Include	b, c, f
S3	226.14.5.2	Exclude	b, c, g
S3	228.24.21.4	Include	d, e, f



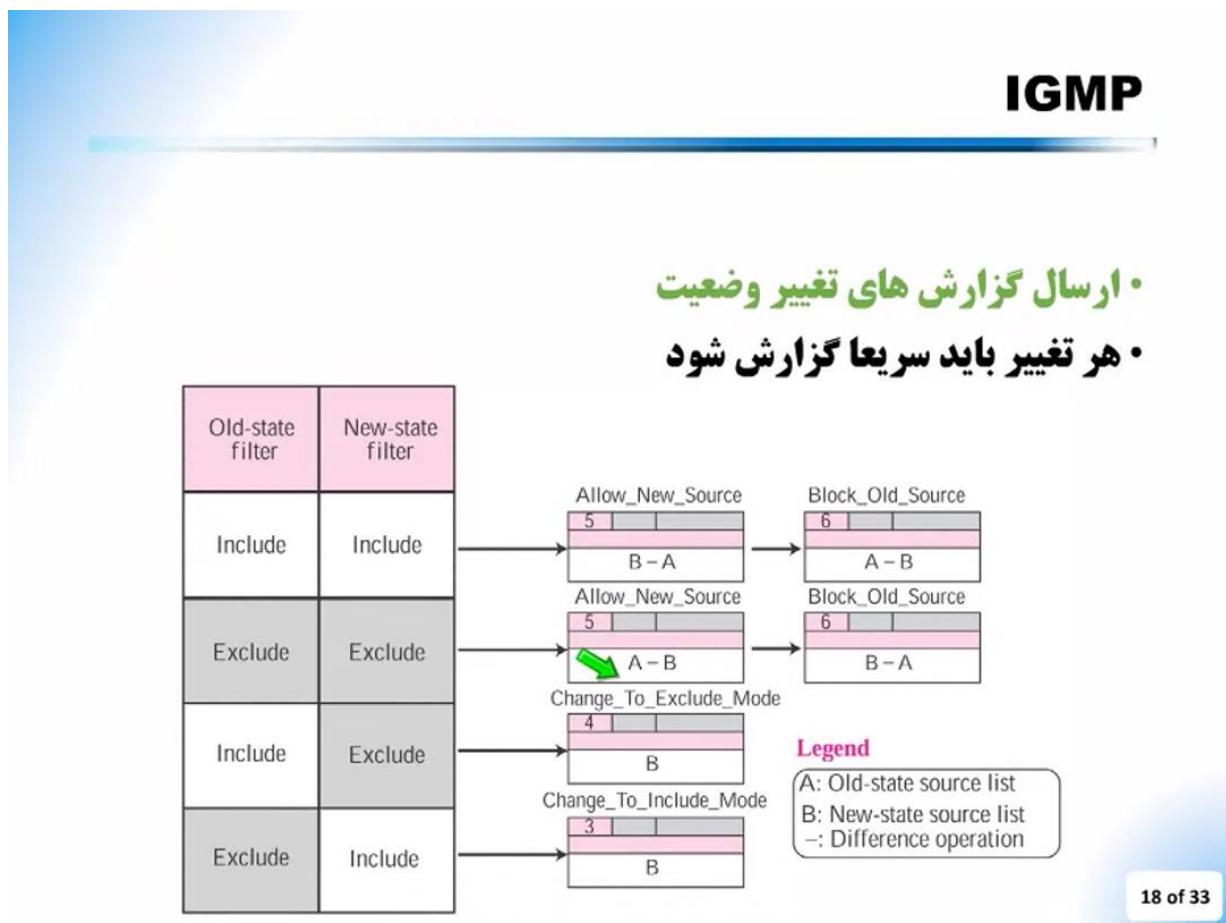
Interface state

Multicast group	Group timer	Filter	Source addresses
226.14.5.2	⌚	Exclude	c
228.24.21.4	⌚	Include	b, c, d, e, f

17 of 33

ممکنه که تغییری اتفاق بیافته توی لیست یک هاست . یک Interface ممکنه تغییری بکنه تو لیستش. این تغییر ممکنه به تغییر وضعیت بیانجام یا فقط سورس ها تغییر کنه . بذارید راجع به این ۴ تغییر که از Include به Exclude که مشخص . از Exclude ، از Exclude به Include و از Include به Exclude مثال بزنم. حالتی که فیلتر تغییر نمیکنه یعنی Exclude و Include ثابت هستند یعنی Exclude به Include و Include به Include که مشخص فقط سورس ها تغییر خواهند کرد مثلا در همین مثال اسلاید قبل اگر ما برای ۲۲۸.۲۴.۲۱.۴ Include از B لیست سوکت ۲ خارج بشه یه تغییر داریم. ولی فیلترمون تغییر نمیکنه. در حالت Include باقی میمونه. اما فرض کنید که یک

اینتری اینجا برای ۲۸.۲۴.۲۱.۴ اضافه بشه و فرض کنید سوکت S1 هم میخواهد به این Join ، Multicast Group رکوردي داشته باشه که فیلترش Exclude باشه در این صورت، بیاین اینو به عنوان یک تمرین همین الان حل کنید. چیز جالبی هم هست. قسمت اول تمرینمون این باشه، سطر اول این جدول همین آدرس Multicast Group ش رو عوض کنید تغییر بدید به ۲۲۸.۲۴.۲۱.۴ و هم فیلتر Source Address را بگنید و لیExclude که این سطر مون چه تغییری میکنه؟ به چی تبدیل میشه؟ در حال حاضر Interface است B,C,D,E,F هم Source Address هستند.



توضیح درباره Include و Exclude : سطر اول و دوم جدول اسلاید ۱۷ را توضیح میدهم تا معنی Include و Exclude را متوجه بشید. میگه Socket ، S<sub>1</sub> میخواهد عضو این گروه هست، گروه با آدرس ۲۲۶.۱۴.۵.۲ و فقط علاقه مند است که پکت هایی از Source های A,B,D,E دریافت کند و میگه من فقط اینها رو میخواهم بهشون Attend یا گوش کنم. S<sub>2</sub> در سطر دوم میگه به این ۲۲۶.۱۴.۵.۲ من Join میشم ولی به A,B,C نمیخواهم گوش بدم . یعنی بسته های از سورس های A,B,C رو نمیخواهم دریافت کنم ولی بسته های دیگه از هرجا بیاد اونارو میخواهم بگیرم. پس فرق Include List و Exclude List در این هست.

درخواست یکی از دانشجویان درباره توضیح درمورد سوال استاد

پاسخ استاد: در اسلاید ۱۶ هم توضیح دادم که اگر تمام Entry های مربوط به یک گروه مثل اینجا تو این مثال ۲۲۸.۲۴.۲۱.۴ فیلترش Include باشه ترکیب اینها با هم یک Include لیست Exclude میکنه. اگر حداقل یک Exclude داشته باشیم مثل ۲۲۶.۱۴.۵.۲ که یدونه ایجاد میکنه. حاصل میشیم که اینجا میبینید. داریم درواقع دوتا Exclude List داریم. حاصل میشیمExclude سطر اولی که اینجا میبینید.

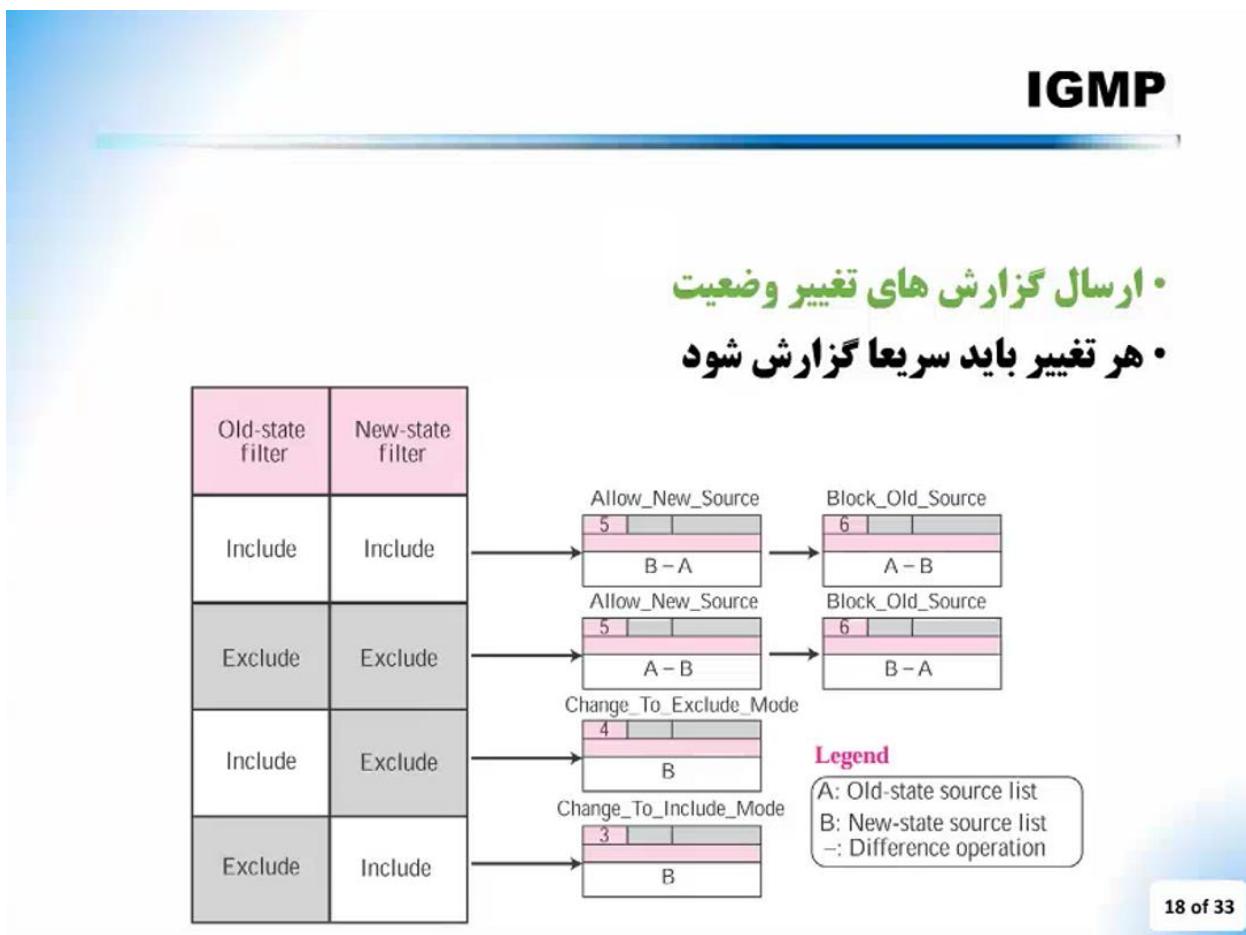
حالا تو این مثال ما برای ۲۲۸.۲۴.۲۱.۴ دوتا Include داشتیم و تمام الان یدونه Exclude میشیم. حداقل یدونه Exclude داریم پس نتیجه میشیم و برای محاسبه اضافه میشیم. باشد یعنی حالت پایینی میشیم که همه Source ها ببینید چکار کردیم؟ در حالتی که همه Include باشند یعنی اجتماع اینها رو در نظر میگیریم اجتماع اون Source Address را رو درنظر میگیریم . و اگر یکسری Include و یکسری Exclude داریم باید اشتراک Exclude ها رو بدست بیاریم

و منهای اجتماع **Include** ها میکنیم. خب دراین مثال ما یدونه **Exclude** داریم پس اشتراکش میشه همون خودش **A,B,D,E** و اجتماع **A,C,D,E,F** ها میشه این تفرق یا اختلاف رو اگر بخواهیم بدست بیاریم **B,D,E** حذف میشه و فقط یه **A** باقی می مونه.

ببینید اشتراک لیست **Exclude** رو اول حساب میکنیم و اختلافش با لیست **Include** رو باید بدست بیاریم. اختلاف حالا فرقی نمیکنه بگید اجتماع **Include** یا بگیم با **Exclude** ها . این تفرق روی تمام آدرس ها باید اعمال بشه. پس اشتراک تمام **Exclude** ها رو **A,B,D,E** حساب میکنیم که توی این مثال یدونه **Exclude** اینجا داریم که هست **B,C,F** اینجا داریم **B,C,F** یعنی از اونجا حذف میشه اختلافش با **Include** ها یه **D,F** که **D,E,F** حذف میشه و فقط **A** باقی میمونه. که **B** حذف میشه و اینکی لیست هم **A** پس لیستمون میشه **Source Address Exclude** .

در اسلاید ۱۷ ما تغییر از **Include** به **Include** رو میتونیم ببینیم مثلا که اگر **A** از اینجا حذف بشه(سطر<sup>۱</sup>) یا فرض کنیم **C** بهش اضافه بشه تغییر از **Include** به **Include** هست. توی این سطر(S<sup>۲</sup>) اگر **A** اضافه بشه تغییر از **Include** به **Include** هست یا اگر یک تغییر در لیستهای **Exclude** اتفاق بیافته تغییر از **Exclude** به **Exclude** هست. اینجا یک تغییر از **Include** به **Exclude** رو دیدم میتونیم تغییر از **Include** به **Exclude** هم داشته باشیم فرض کنید اگر فرض کنید این دوتا سطر ما حذف بشه یعنی دیگه سوکت ۲ و S<sup>۳</sup>

کلا بخوان از این ۲۲۶.۱۴.۵.۲ Multicast Group جدا بشن یعنی خارج بشن از عضویت این گروه خارج بشن سوکت S۲ و S۳ . فرض کنید اپلیکیشن بسته شده. دیگه این سوکت آزاد میشه و دیگه عضو این گروه هم نخواهد بود. در این صورت ما فقط یک لیست داریم اینم این لیست سطر اول که Include پسExclude ما تو سطر اول تبدیل میشه به . پس تغییر از Exclude به Include و Exclude به Include و Exclude به Include و Include به Include رو می تونیم داشته باشیم.



اسلاید ۱۸ - پس در هر حالتی ببینیم چه لیست هایی باید Advertise یا گزارش بشه. کدوما باید Report بشه. در حالت Include به Include State یعنی قدمی مون Include و Source جدیدمون فیلترش همچنان Include یعنی تغییر فقط در نودهای State میخواهیم ازشون پکت دریافت بکنیم یعنی علاقمندیم دریافت بکنیم تغییر کرده. خب این دو حالت داره یا یه سورس اضافه شده یا یک سورس کم شده حذف شده . اگر سورسی اضافه شده باشه، تایپ پکتمون که گزارش رو میفرستیم، پکت ریپورتمون ۵ خواهد بود و لیستی که گزارش میکنیم B-A هست یعنی همون نودی که یا سورس آدرسی که اضافه شده. B لیست سورس هایی State جدیدمون و A هم لیست سورس های State قدیمی مون هست . خب هم که میدیم میشه A-B یعنی همون سورس هایی که حذف شدند. باز برگردم به این مثال (اسلاید ۱۷) فرض کنید اینجا توی این Include List یه A اضافه شده ماباید پکت با تایپ ۵ و سورس آدرس A ارسال بکنیم . گزارشمون به این شکل خواهد بود. اگر فرض کنید که B حذف شده یعنی B اول بوده الان دیگه نیست A-B یعنی لیست A منهی لیست B میشه همون سورس B . یا بهتره مثال دیگه ای بزنیم غیر A و B . فرض کنید F حذف شده در اینصورت A-B میشه F . و تایپ پکتمون میشه ۶ . درواقع تایپ ۵ میگه که من میخوام به این سورس ها هم گوش بدم یعنی پکت هاشون رو دریافت کنم ، تایپ ۶۶ میگه من دیگه نمی خوام به این سورس گوش بدم . یا پکتشو دریافت کنم.

خب حالا Exclude یعنی چی؟ یعنی میگه که Exclude میگه من نمی خوام به اینا گوش بدم به بقیه می خوام گوش بدم. فرض کنید لیست Exclude ما شامل B,D,F باشه می گه نمی خوام از اینا پکت بگیرم از بقیه می خوام بگیرم . حالا فرض کنید که یه سورس دیگه اضافه بشه به این لیست معنی چیه؟ ببین Allow\_New\_Source یعنی میخوام به یه سورس جدید هم گوش بدم. یعنی اون سورسی که می خوام بهش گوش بدم از اون Exclude List من خارج شده یعنی اون سورسی که می خوام بهش گوش بدم از اون Exclude List بوده الان میخوام به F نمیخواستم گوش بدم الان میخوام گوش بدم. پس F رو باید از این Exclude List خارج کنم الان لیستم میشه B,D و از A-B در میاد. پس اختلاف رو چجوری باید بگیرم بعد گوش بدم. پس پکت با تایپ ۵ میفرسته و با لیست A-B و برعکس این یک سورس دیگه رو بلاک کنیم نخواهیم به پکت هاش گوش بدیم باید به اون رو اضافه کنیم. و B-A میشه همون سورسی که درواقع بلاکش کردیم از این به بعد و پکت رو از تایپ ۶ می فرسته.

خب حالا State مون اگر فیلترش از Exclude تغییر بکنه دیگه راحته این. فیلتر State مون از Exclude به تغییر بکنه پکت تایپ ۴ رو میده تایپ ۴ یعنی این من هست و B یعنی لیست جدید من هم بشکونه به روتр در واقع این گزارش ها به روتر در واقع ، هاست به روتر این گزارش رو میده. به روتر میگه من از الان به بعد به این سورس ها نمیخوام گوش بدم به بقیه می خوام گوش بدم و اگر فیلتر ما از Exclude به

تغییر بکنه بازم لیست B رو باید بدیم و لیست جدید رو درواقع و پکت تایپ ۳ که میگه من فقط به اینا می خوام گوش بدم و به بقیه نمی خوام گوش بدم. این پس تغییر فیلتر و Include به Exclude ، Exclude به Exclude ، Include به Include لیست ها از Include به Exclude دیدین که چه رفتاری باید باهاش بکنید و چه گزارشاتی باید به روتر بدهید.

درخواست یکی از دوستان لطفا Exclude را مجدد توضیح بدهید.

پاسخ استاد : یه لیست Exclude داریم B,D,E میگیم نمیخوام از اینا پکت دریافت کنم الان میخوام E رو از این به بعد دیگه پکت دریافت بکنم برای اینکه بخوام از E پکت دریافت بکنم سورس E رو باید از Exclude List خارج بکنم . الان B,D,E م بوده Exclude List شده B,D . حالا چجوری باید به روتر بگم ؟ باید به روتر بگم من از این به بعد می خوام به این سورس هم گوش بدم پس پکت های این سورس رو هم میخوام دریافت بکنم نوع پکتش بود پکت نوع ۵ و لیست هم میشه A-B یعنی A اون لیست بزرگه حالا یه چیزی هم ازش کم شده اختلافش میشه اون سورس هایی که از این به بعد می خواهیم بهش گوش کنیم. برعکسش هم هست حالا اگر بخوام یه سورس دیگه علاوه بر سورس های دیگه هم گوش بدم یه سورس دیگه هم بلاک کنم لیستم یکم دیگه بزرگتر میشه مثلا B,D,E بوده F رو هم بهش اضافه کنم. میخوام بگم اونی که جدیدا اضافه شده رو بلاک کن. تایپ پکت ۶ و اونی که جدید اضافه شده میشه A-B . B-A منهی B,D,E . F میشه F رو بلاک کن.

## ۰ دریافت گزارش های پرس و جو

۰ با دریافت پرسش بلا فاصله پاسخ داده نمی شود

۰ یک زمان تصادفی که بر اساس بخش **Max Resp Code** محاسبه شده صبر می کند

۰ عملیات میزبان وابسته به نوع پرسش است:

۰ پرسش عمومی : مقدار دهی تایمر واسطه به مقدار محاسبه شده. لغو تایمر قبلی

۰ پرسش وابسته به گروه : تایمر گروه به کمترین مقدار بین مقدار قبلی و مقدار محاسبه شده تنظیم می شود. تایمر کار نمی کرده : بی نهایت

۰ پرسش وابسته به گروه و مبدأ : مشابه قبلی. بیست مبداها برای پاسخ با تأخیر حفظ می شود

19 of 33

اسلاید ۱۹ - چند تا نکته اینجا هست . بعد از اینکه کوئری رو گرفت، درواقع کوئری رو کی میده ؟ روترا و هاست اون کوئری رو میگیره و به محض اینکه این کوئری رو گرفت Replay نمیکنه و ریپورت رو درواقع نمی فرسته. یه زمانی به صورت تصادفی براساس اون کدی که توی فیلد Max Resp Code هست صبر میکنه براساس اون محاسبه می شه این زمان تصادفی و بعد این میزان صبر میکنه بعد اون Replay یا ریپورت رو ارسال میکنه. کارهایی که هاست می کنه براساس نوع کوئری هست. اگر کوئری جنرال باشه (روی جدول پایین اسلاید ۱۷) بین هرکدام از این Entry ها یدونه تایمر دارم بهش میگیم Group Timer تایمر گروه. این تایمر ها کاربرد دارند که دربارش صحبت میکنیم. و کل Interface هم یه تایمر داره که راجع بهش صحبت میکنیم.

خب اگر جنرال کوئری دریافت بکنه هاست Interface Timer رو به مقدار محاسبه شده Set میکنه یعنی اون مقداری که از Max Resp Code محاسبه کرده و تایمر قبلی که داشتیم Set لغو ش میکنه و تایمر رو به مقدار محاسبه شده Set میکنه. تایمر رو به مقدار محاسبه شده میکنه.

درمورد کوئری های وابسته به گروه تایمر گروه رو تغییر میده. چه تغییری؟ میاد کمترین مقدار بین اون مقداری که قبلا داشته و مقدار محاسبه شده رو بدست میاره . مثلًا تایمرمون مقدار قبلش بوده ۱۲ الان محاسبه کردیم شده ۶ . کمترینش ۶ پس تایمر گروه رو ۶ انتخاب میکنیم. مثلًا قبل بوده ۱۲ الان محاسبه کردیم شده ۱۷ پس ۱۲ کوچیکتره مقدار رو به ۱۲ Set میکنیم.

اگر تایمر گروه کار نمیکرده یعنی قبلا Set نشده بوده اون مقدارش بی نهایت و مقدار محاسبه شده قطعا از اون بی نهایت کمتر و معنیش اینه اگر تایمر کار نمیکرد اون مقدار محاسبه شده رو جاش میداریم توی تایمر گروه میداریم.

و اگر کوئری وابسته به گروه و مبدا هست مثل حالت قبلی کوئری وابسته به گروه اقدام میکنیم و لیست مبدا رو برای پاسخ با تاخیر حفظ میشه که توی کوئری هست رو نگه میداریم که وقتی تایمر منقضی شد برای اونها پاسخ بدیم.

پس در مورد این تایمرها که گفتیم یک تایمر گروه داریم Group Timer و یک تایمر واسط داریم Interface Timer بسته به اینمه چه کوئری ای میاد فرق میکنه که هاست چیکار میکنه. هاست این کارهایی که گفتیم رو انجام میده .

Type 1 : Mode-Is-Include  
Type 2 : Mode-Is-Exclude

### ۰ انقضای تایمر

- ۰ ارسال پیام های گزارش عضویت بعد از انقضای یک تایمر
- ۰ نوع و تعداد رکوردها وابسته به تایمر
- ۰ ۱- تایمر واسط بعد از پرسش عمومی : ارسال یک گزارش عضویت شامل یک رکورد وضعیت فعلی به ازای هر گروه در وضعیت واسط
- ۰ ۲- تایمر گروه بعد از پرسش وابسته به گروه : ارسال یک گزارش عضویت شامل یک رکورد وضعیت برای گروه مورد نظر
- ۰ ۳- تایمر گروه بعد از پرسش وابسته به گروه و مبدأ : ارسال یک گزارش عضویت شامل یک رکورد وضعیت برای گروه مورد نظر

20 of 33

اسلاید ۲۰ - اگر تایمر منقضی بشه یعنی مثلا گفتیم ۲ ثانیه و ۲ ثانیه تموم بشه یا مثلا ۱۵ ثانیه و ۱۵ ثانیه تموم بشه. منقضی بشه تموم بشه تایمری که کار میکرده. پیام های این گزار عضوریت یا در واقع اون ریپورت هایی که گفتیم. ریپورت مسیح ها بعد از اینکه تایمر منقضی شد ارسال میشه. نوع و تعداد رکوردهایی که تو این پیام گزارش عضویت قرار میگیره وابسطه به تایمر هست. این ۳ حالت ممکنه بوجود بیاد.

تایمر واسط بعد از یک پرسش عمومی یا جنرال کوئری منقضی شده در این صورت ارسال گزارش عضویت شامل یک رکورد وضعیت فعلی به ازای هر گروه در وضعیت واسط هست یعنی ما گزارش عضویتمون شامل رکوردهایی هست که توی اینترفیس هست. رکوردهای توی اینترفیس رو توی

اسلاید ۱۷ براتون مثال زدم. رکوردهای جدول پایینش رو میگیم رکوردهای اینترفیس. همونایی که از ترکیب State های جدول بالای همون اسلاید بدست میاد. State هایی که مربوط به Socket هاست. ترکیب اونها یکسری رکورد میده برای اینترفیسمون. حالا بعد از منقضی شدن تایمر واسط، بعد از یک جنرال کوئری یک گزارش عضویت ارسال میشه که این گزارش عضویت یکسری رکورد دارد . این رکورد ها همان رکوردهای وضعیت هرگروه در اینترفیسمون. اگر کوئریمون وابسطه به گروه باشه و بعد تایمر گروه منقضی بشه اونوقت یک گزارش عضویت شامل یک رکورد وضعیت برای اون گروه ارسال میشه. گروه آدرس داره دیگه فرض کنید داشتیم ۲۲۴.۲۴.۱۲.۴ به طور مثال، این گروهی که براش کوئری اومنده اگر تایمرش منقضی بشه گزارش براش ارسال میشه . و اگر تایمر گروه پس از پرسش وابسطه به گروه و مبدا منقضی بشه دقیقا مثل حال قبل یک گزارش عضویت شامل یک رکورد وضعیت برای گروه مورد نظر ارسال میشه.

## IGMP

- پروتکل IGMP در مسیریاب ها
- مسیریاب چندپیغشی = پرسشگر (querier)
- نیاز به مدیریت ۶ نوع رکورد گروه

اسلاید ۲۱ – خب بريم ببینيم تو مسیریاب ها پروتکل Igmp چه تاثیری ميذاره. توی مسیریاب ها چه اتفاقی می افته. گفتیم اون کسی که کوئری رو میده کیه؟ مسیریاب هست. در واقع یه مسیریاب Multicast Enable Router یا Multicast Router Multicast کوئری میده،

کوئری رو به Host ها میده تا متوجه باشه که هر کدوم از این هاست‌ها عضو چه گروه‌هایی هستند. و در واقع به کدوم Multicast Address‌ها گوش میدن و باید بسته‌ما به اون مقصد ارسال بشه. توی روتر مدیریتش نیاز به ۶ نوع رکورد گروه داریم که راجع بهش می‌خواهیم صحبت بکنیم.

## IGMP

- وضعیت پرسشگر
- نگهداری اطلاعات وضعیت برای هر گروه چندپخشی مرتبط با هر واسط
- جدولی که هر سطرش مربوط به یک گروه چندپخشی است
- اطلاعات: آدرس چندپخشی، تایمر گروه، حالت فیلتر و رکوردهای مبدأ
- هر رکورد مبدأ: مبدأ و تایمر

State for interface m1				
Multicast group	Timer	Filter	Source addresses	
227.12.15.21	⌚	Exclude	(a, ⚡)	(c, ⚡)
228.21.25.41	⌚	Include	(b, ⚡)	(d, ⚡) (e, ⚡)
State for interface m2				
Multicast group	Timer	Filter	Source addresses	
226.10.11.8	⌚	Exclude	(b, ⚡)	
227.21.25.41	⌚	Include	(a, ⚡) (b, ⚡) (c, ⚡)	
228.32.12.40	⌚	Include	(d, ⚡) (e, ⚡) (f, ⚡)	

22 of 33

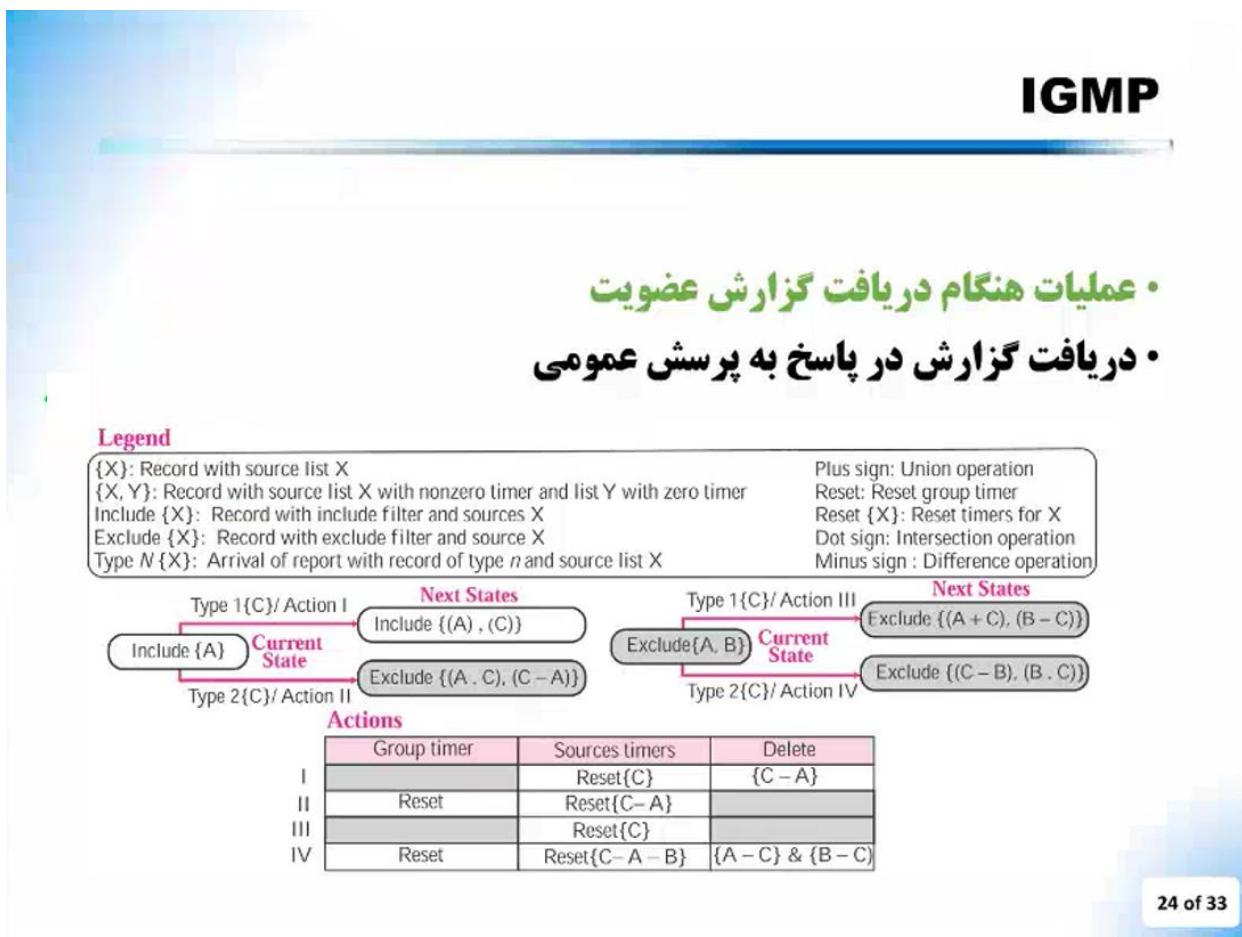
اسلاید ۲۲ - اطلاعات وضعیت برای هر گروه Multicast مرتبط با هر اینترفیس روتر باید نگهداری بشه مثلاینجا یه روتر داریم یه اینترفیس M<sup>1</sup> برای هر کدوم M<sup>2</sup> برای اینترفیس M<sup>1</sup> داریم یه اینترفیس M<sup>2</sup> برای هر گروه Multicast نگهداری بشه مثلاینجا دوتا جدول از اینها یک اطلاعات وضعیت باید برای هر گروه Multicast مرتبط با هر گروه M<sup>1</sup> هست این بالا و یکی State اینترفیس M<sup>2</sup> هست میبینید دیگه. یکی State اینترفیس M<sup>1</sup> هست این بالا و یکی State اینترفیس M<sup>2</sup> هست

این پایین. توی اینجا هم می بینید آدرس Multicast داریم یه تایمر برای هر کدوم از اینا داریم، فیلتر ور داریم که Include و Exclude میتونه باشه و بعد سورس ها رو داریم که میتونه به ازای هر سورس هم یه تایمر اینجا داریم. پس توی روتر علاوه براینکه تایمر گروه رو داریم تایمر Multicast رو هم داریم. توی این جدولی که میبینید هر سطرش مربوط به یک گروه است که آدرس گروهش توی یکی از فیلدها قرار داره. اطلاعات که اینجا هست، آدرس Include که گفتم، تایмер گروه، فیلترمون هست که Exclude یا Multicast هست و رکوردهای مبدا که هر رکورد مبدا شامل مبدا و یه تایمر به ازای اون هست.



اسلاید ۲۳ - وقتی یه پرسش جنرال یا یک کوئری جنرال رو روتر ارسال کرده در پاسخ به اون حالا بعد از انقضای تایمر مربوطه یک گزارش دریافت میکنه یک رکورد دریافت می کنه. گزارش هایی که دریافت میکنه شامل رکورد وضعیت فعلی Host هست یا اون اینترفیس مربوط به Host ی که پاسخ داده. با دریافت این گزارش، گزارش رو که روتر دریافت میکنه وضعیت

داخلش ممکنه تغییر بکنه . اینجا فرض کنید که یه وضعیتی اینجا داریم مثلا برای ۲۲۷.۱۲.۱۵.۲۱ ما یک فیلتر Exclude با سورس های A,C . حالا ممکنه یه کوئری جنرال که میزنيم اين سطر تغیير بکنه مثلا A حذف بشه به عنوان مثال يا D اضافه بشه. يع تغیيری ممکنه اضافه بشه ممکن هم هست هیچ تغییری اتفاق نیافته و در همین وضعیت بمونه. پس بعد از دریافت گزارش وضعیت مسیریاب ممکنه که تغییر بکنه.



اسلاید ۲۴ - ببینید در دریافت یک گزارش در پاسخ به یک جنرال کوئری چه اکشن هایی در واقع روترا باید انجام بده؟

این جدول یکم نیاز به تفسیر دارد. یکم باید صحبت بکنم راجع بهش . علائمی که توی این State Source List میبینید توی این قسمت تعریف شده. مثلا {X} میگه Transition Diagram . مثلا اینجا {A} میگه ما یه لیست A داریم و فیلترش هم Include X هست.

X,Y یعنی چی ؟ یعنی یه رکوردی با X و تایمر غیر ۰ و لیست Y با تایمر ۰ . یعنی یه سری از اینها تایمرشون غیر صفر یه سری صفر.

Source X و Include رکورد با فیلتر Include{X}

مشخصه Exclude

دریافت یک رکورد با رکورد Type N و با X Type N {X}

علامت جمع یعنی عملیات اجتماع

به معنی ریست کردن تایمر گروه پس {X} ریست کردن تایمر گروه برای لیست Reset . X هست.

علامت نقطه اشتراک هست و علامت تفریق هم اختلاف دوتا لیست هست.

حالا ببینیم اگر یک Include Record ای داریم یا یک Include List ای داریم مجموعه A یعنی یه رکوردی توی وضعیت روتر داریم شامل سورس های D,E,F مثلا فرض کنید که مجموعه A برابر است با سورس ها D,E,F و فیلترش هم Include هست.

اگر رکورد با تایپ ۱ بیاد و State (Type ۱ {C}), List C میمونه و Include {{(A),(C)}} میشه و میشه {A} که سورس های A تایمر لیستموم C بهش Concat دارند و سورس های C تایمر Zero.

اگر بیاد اکشن اینه. اشتراک A و C تایمر Nonzero میمونه و اختلاف C,A و لیست Type ۲ هم به Incloud تغییر میکنه.

ببینیم تایج ۱ و ۲ چیه که چنین تغییری ایجاد میکنه.

(اسلاید ۲۰) اینجا گفتم. گزارش عضویت در پاسخ به پرسش عمومی یا از تایپ ۱ میتونه باشه یا تایپ ۲. تایپ ۱ یعنی Include Mode، تایپ ۲ یعنی Exclude Mode. یعنی یه لیستی میده و میگه تایپ یک میگه این و تایپ ۲ میگه Exclude List.

برگشت به اسلاید ۲۴- خب وقتی بیایم اینجا متوجه میشیم. پس میگیم اینجا یه داشتیم با اعضای مجموعه A و یکسری Include List دیگه هم بهش اضافه میش. تایپ ۱ Include List C بود دیگه اینجا تو جدول اضافه میشه تایمرشون . قبل تایمرشون داشته کار میکرده ولی اینایی که جدید الان اضافه میشه تایمرشون . هست و ۱ Action هم اتفاق میافته که اینجا تو جدول گفته. اکشن ۱ میگه سورس تایمر ها ریست بشن برای گروه C-A و C هم دیلیت یا حذف بشه.

اختلاف C-A میشه اونایی که توی C هستند و توی A نیستند. مثلا فرض کنید که A شامل B,C,D باشند و B هم شامل E,D باشند . میشه . بزارید اکشن های جدول رو ببینیم . من

بررسی کنم احساس میکنم باید اشتراکشون باید از  $C\{A\}$  حذف بشه چون اشتراک  $C$  و  $A$  باید از  $C\{A\}$  حذف بشه. وقتی از  $C\{A\}$  حذف بشه تایمرش ریست میشه . یعنی اونایی که تازه ریپورت برashون اومنده مثلًا توی این مثال  $C,D,E$  گفتیم و  $D,F$  رو . برashون ریپورت جدید اومنده و  $D,F$  تایمرشون باید . بشه و ان  $D$  که مشترک بود توشنون باید از  $A$  حذف بشه. به نظرم این اتفاق باید بیافته.

حالا اگر لیستمون  $Include$  باشه و  $\exists$  یعنی  $Type$  بیاد گفتیم ترکیب  $Include$  و  $Exclude$  قطعاً میشه  $Include$  واعضاً میشن اشتراک  $A,C$  و اختلاف  $C,A$  که اشتراک  $A,C$  تایمرشون باقی میمونه . اختلاف  $C,A$  هم تایمر . خواهند داشت.

در این حالت  $Group Timer$  ریست میشه چون  $Include$  List مون تبدیل شده به  $C-A$  در واقع تغییر اساسی کرده و تایمر  $C-A$  ریست میشه و چیزی حذف نمیشه. اگر  $Exclude$  List داشته باشیم این  $A,B$  معنیش همون چیزیه که در واقع میبینید. اگر  $A$   $Exclude$  List داشته باشیم شامل سورس های مجموعه  $A$  و مجموعه  $B$  که مجموعه  $B$  در این صورت  $Exclude$  باقی میمونه چون  $Include$  و تایمرش غیر صفر هست و مجموعه  $B$  تایمرش صفر هست ریپورت تایپ ۱ میاد یعنی  $Include$  تایمر ش میشه  $\exists$  . ریست میشه. و درصورتیکه  $Exclude$  ریپورت بیاد اینشکلی اعضا تغییر

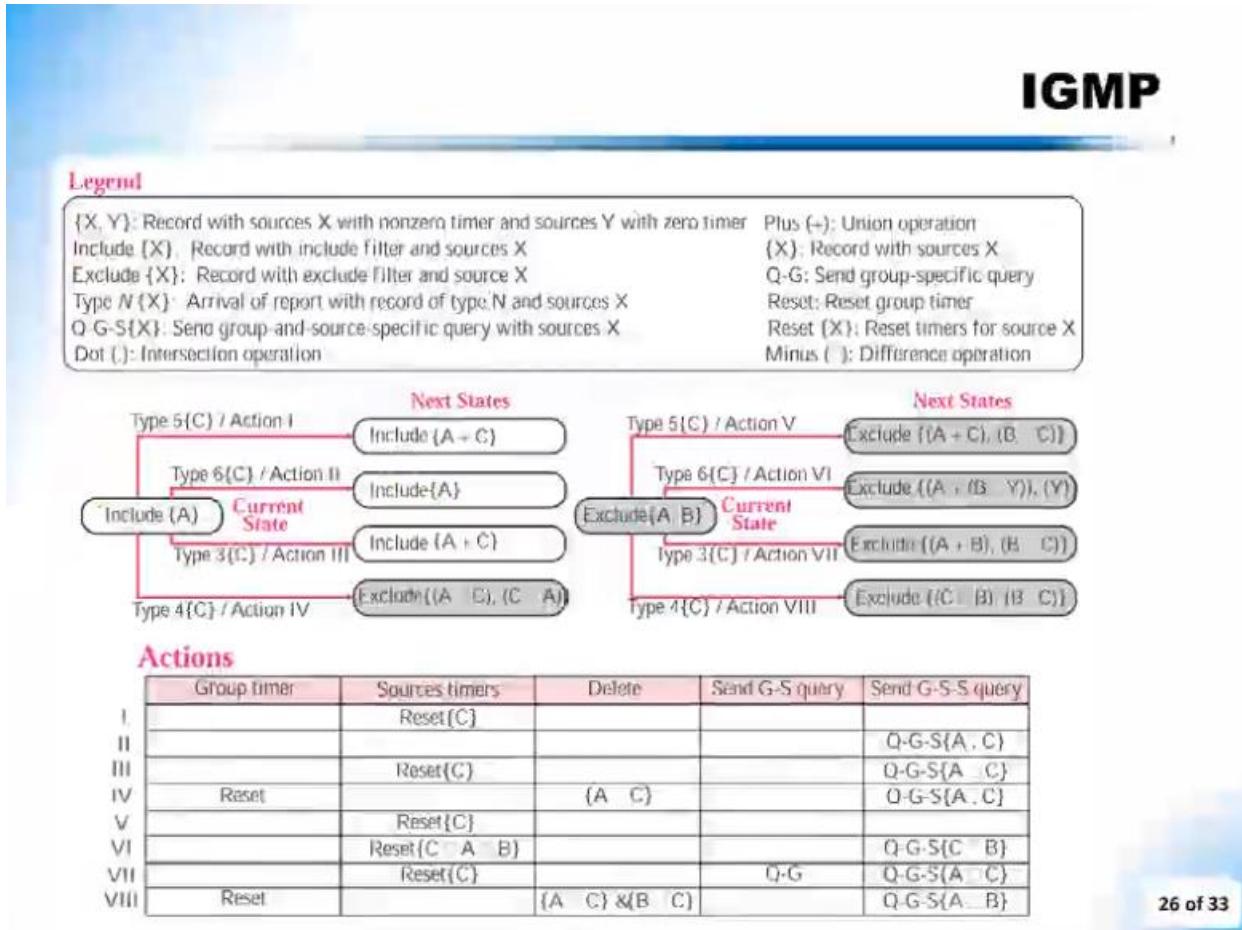
میکنه . باقی میمونه C-B تایمرش . میشه و B اشتراکش با C تایمرش .  
میشه . یعنی تو حالت بالا اکشن ۳ میبینید C تایمرش ریست میشه و توی اکشن ۴ گروپ تایمر  
Rیست میشه A-B هم تایمرش Rیست میشه . و (A-C) & (B-C) یعنی C هم دلیت میشه.

این State Transition Diagram ی که تو اینجا توی Rfc هست رو نگاه کردم این Rfc قطعا مشکل داره . دقیقا اشکالاتی توی این وجود داره که مطابق Transition Diagram نیست . من شکل Rfc رو آخر کلاس بهتون نشون میدم. یه چیزاییشو متوجه نشدم . قبل دقت نکرده بودم که مثلا فرض کنید توی Exclude List احتمالا دلیلی داره ولی چرا A نیست که قبل بوده توی Exclude List بوده مثل A هست . و سورس B,C,D هست و D هست  
که این D تایمرشون غیر صفر و H,G تایمرشون . هست . وقتی یک G,H  
Mیاد مثل H,K در اینصورت چرا توی Exclude List ی که توی نکست استیت داریم C هیچکدوم از اعضای A نیستند . باید اختلاف C,D هست و اشتراک B,C اینو هنوز درک نکردم و ققلا هم بهش فکر نکرده بودم حقیقتش فقط دیده بودم اتفاق می افته و حتی تو Rfc هم ننوشته چرا این اتفاق افتاده . ولی قطعا این State Transition Diagram که از کتاب برداشتم از کتاب فروزان هست قطعا اشکال داره . واضحترین اشکالش این دلیت ۱ اکشن C-A توی اکشن ۱ نباید باشه . این توی Rfc توی اکشنی هست که A داریم و B Exclude می اره به عنوان ریپورت مثل . یعنی این سطر باید اینجا تو سطر ۳ باشه . خیلی اهمیت نداره . ما هدفمون فوکوس

کردن روی پروتکل Igmp خصوصاً توی روتر که نیست بنا براین این رو همینجوری قبول نکنید.  
اگر علاقه دارید Rfc ۳۳۷۶ همین Igmp هست.



اسلاید ۲۵ - موقع دریافت گزارش در پاسخ به پرسش‌های دیگه غیر از جنرال کوئری گفتیم تایپ ۳ و ۴ و ۵ و ۶ میاد. در تایپ ۳ و ۴ Filter Mode مون تغییر کرده یعنی از Include به رفته و یا از Exclude میگه می خواه از این سورس هم دیتا دریافت کنم . ۶ میگه از این سورس دیگه دیتا دریافت نکنم.



اسلاید ۲۶ – تغییر وضعیت مثل این اسلاید است. اگر لیستمنون **Include** باشد و تایپ ۵ بیاد تغییر به این شکل اتفاق می افته . تایپ ۶ بیاد به این شکل و تایپ ۳ و ۴ هم مطابق اسلاید پایین.

و اگر لیستمنون **Exclude** باشد تایپ ها مطابق اسلاید قابل مشاهده است. اکشن ها هم تو جدول نوشته و باخوندن **State Transition Diagram** وجود آشنا شدید. برید بررسی کنید. احتمال داره اینم با Rfc یکسان نباشد و اشکال باشد یا منطقشو درک نکنیم. خیلی مهم نیست فقط می خوایم بگیم این اتفاق توی Igmp روی روتر های مالتی کست می افته.

## • نقش IGMP در جلورانی

### • توصیه های جلورانی:

- در نسخه های قبل فقط بر اساس آدرس گروه
- در نسخه ۳ بر اساس آدرس گروه و مبدأ

### • شش توصیه جلورانی

Filter Mode	Source Address	Source Timer Value	Recommendation
Include	In the list	greater than zero	Forward
Include	In the list	zero	Do not forward
Include	Not in the list		Do not forward
Exclude	In the list	greater than zero	Forward
Exclude	In the list	zero	Do not forward
Exclude	Not in the list		Forward

27 of 33

اسلاید ۲۷ - یه سری توصیه داره Igmp برای Forwarding طبق این نسخه ۳ انجام بشه توی نسخه های قبل از نسخه ۳ (آخرین نسخه ۳ هست). یعنی نسخه ۱ و ۲ فقط براساس آدرس گروه و توی نسخه ۳ براساس آدرس گروه و مبدأ چون تو نسخه ۳ هست که بحث مبدا اضافه شده. ۶ تا Recommendation اینجا داریم. اگر Filter Mode Include باشه و Source Address داخل لیست باشه یعنی و Forward هم بزرگتر از ۳ باشه میگه Source Timer کن. اگر Source Address باشه و Source Timer هم باشد . میگه Forward نکن.

اگر باشه Forward نکن و به Source Address داخل لیست نباشه و میگه Include هم کاری نداره.

اگر هم Source Timer داخل لیست باشه یعنی و Source Address باشه و Exclude بزرگتر از ۳ باشه میگه Forward کن.

اگر هم Source Timer داخل لیست باشه و Source Address باشه و Exclude میگه Forward نکن.

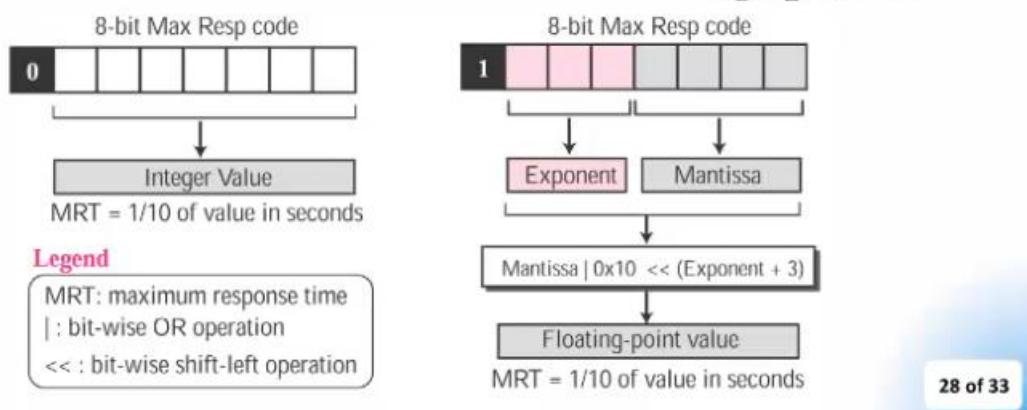
اگر باشه Forward کن و به Source Address داخل لیست نباشه و میگه Exclude هم کاری نداره.

این Recommendation برای Igmp هایی هست که برای روتر داره و روتر هم براین اساس عمل میکنه.

• متغیرها و تایمرها

• حداکثر زمان پاسخ

- حداکثر زمان مجاز قبل از ارسال یک گزارش در پاسخ به یک پرسش
- محاسبه زمان :



اسلاید ۲۸ – محاسبه حداکثر زمان مجاز قبل از ارسال یه گزارش در پاسخ به یک پرسش به این شکل محاسبه میشه. اگر فیلد ۸ بیتی Max Res Code ارزشش . باشه این Integer Value

۱ دهمش میشه Mrt . در ثانیه

اینجا هم Legend رو گفته. (مطابق اسلاید)

مثلا اگر ۴۰ باشه یک دهمش میشه ۴ و میشه ۴ ثانیه . رندوم بین . تا ۴ ثانیه میشه.

و اگر بیت ارزشش ۱ باشد این یه عدد اعشاری Exponent و Mantissa هست و اینجوری Floating Point بعدی میاد و Mrt میشه یک دهم این Floating Point عمل میشه و یه Value بزرگتر از ۱۲۸ . بدست اومند.



اسلاید ۲۹ - فیلد Qrv یعنی Query Robustness ، Robustness ، Variable در واقع Response میکنه که هاست چند مرتبه بسته Variables رو بده که مطمئن بشیم دست روتر میرسه.

## IGMP

- متغیرها و تایمراه
- بازه پرسش پرسشگر
- بازه بین پرسش های عمومی.
- مقدار پیش فرض = ۱۲۵

اسلاید ۳۰ – بازه پرسش Querier که بازه کوئری های عمومی چقدر باشه مقدار پی فرضش هم ۱۲۵ هست.

## IGMP

- جاسازی (encapsulation)
- جاسازی بسته های IGMP در بسته های IP
- بخش پروتکل = ۲
- بخش TTL = ۱
- IP مقصد وابسته به نوع پیام

Message Type	IP Address
General Query	224.0.0.1
Other Queries	Group address
Report	224.0.0.22

اسلاید ۳۱ - بسته های Igmp در بسته های Ip قرار میگیره. بخش پروتوكلش میشه ۲ و بخش Ttl هم ۱ هست و مقصد وابسته به نوع پیام هست.

اگر باشه میشه این و اگر وئری های دیگه باشه گروه آدرس اینجا قرار میگیره و ریپورت هم باشه میشه این آدرس.



- سازگاری با نسخه های قبلی
- نسخه ۳ پیام های نسخه های ۱ و ۲ را دریافت می کند

Version	Type Value	Message Type
1	0x11	Query
	0x12	Membership Report
2	0x11	Query
	0x16	Membership Report
	0x17	Leave Group

32 of 33

اسلاید ۳۲ - نسخه ۳ با نسخه های ۱ و ۲ سازگاری داره و پیام رو دریافت میکند.

## جلسه یازدهم:

# پروتکل پیام های کنترلی در اینترنت

کنترل خطای اینترنت  
**ICMP**

## ICMP

- عدم وجود سازوکار گزارش یا تصحیح خطای اینترنت در پروتکل IP
- خطاهای احتمالی مسیریابی
  - مسیر اشتباه
  - عدم پیدا کردن گام بعدی
  - صفر شدن فیلد TTL
- حذف همه بخش های یک پیام قطعه قطعه شده بدلیل انقضای زمان و عدم دریافت بخشی از پیام
- عدم وجود سازوکار پرس و جو در مورد میزبان ها و پرس و جوی مدیریتی در پروتکل IP
- **ICMP**: رفع کمبودهای IP در زمینه خطاهای مسیریابی

در پرتفکل IP ساز و کار گزارش تصحیح خطای وجود ندارد ولی در پرتفکل های لایه بالاتر مثل TCP بحث کنترل خطای داریم. در نظر بگیرید که خطای صرفا تصحیح بسته نیست.

خطاهای احتمالی مسیر یابی که در لایه ۳ ممکن است اتفاق بیفتد (خطاهای پر تکرار) به ترتیب در اسلاید ۴ لیست شده به شرح زیر می باشد:

**مسیر اشتباه:** به عنوان مثال بسته به روتور میرسد که مسیر به سمت مقصدش به پورتی هست که بسته را از آن دریافت کرده

عدم پیدا کردن گام بعدی: روتور هیچ مسیری به سمت اون مقصد بسته ندارد

صفر شدن فیلد TTL : فیلد (Time to live) وقتی صفر شود روتور بسته را حذف میکند

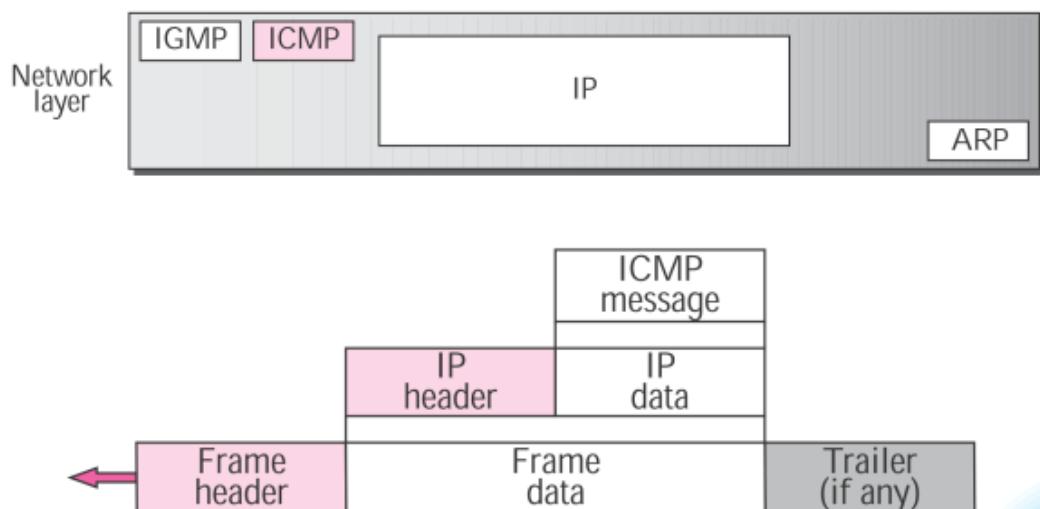
حذف همه بخش های یک پیام قطعه قطعه شده به دلیل انقضای زمان و عدم دریافت بخشی از پیام: حذف همه بسته های پیام به دلیل نداشتن بخش هایی از پیام دریافتی یا زمان آن فرگمنت منقضی شده است. که این خطای نیز باید گزارش شود.

**عدم وجود سازوکار پرس و جو در مورد میزبانها و پرس و جوی مدیریتی در پرتفکل IP**

به دلیل کمبودهایی که در پرتفکل IP وجود دارد پرتفکل ICMP یا کنترل خطای اینترنت معرفی شد: رفع کمبودهای IP در زمینه خطاهای مسیریابی

# ICMP

## • جایگاه ICMP :



5 of 21

در تصویر جایگاه پرتلک ICMP در کنار IGMP و نوع بسته بندی پیام های آن در پایین شکل گویا می باشد.

- پیام های ICMP :
- پیام های گزارش خطا
- پیام های پرس و جو (query)

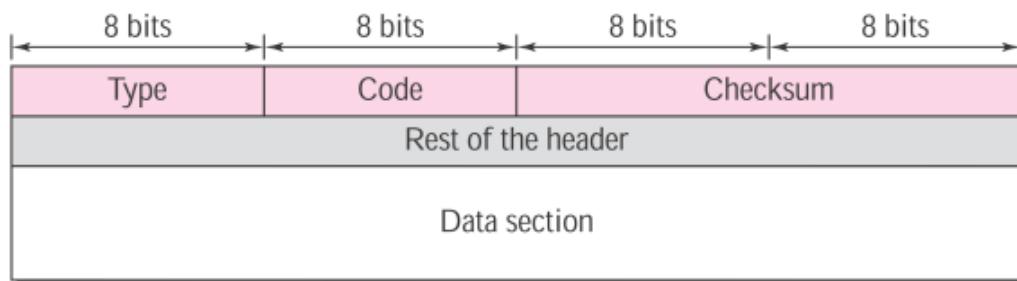
Category	Type	Message
Error-reporting messages	3	Destination unreachable
	4	Source quench
	11	Time exceeded
	12	Parameter problem
	5	Redirection
Query messages	8 or 0	Echo request or reply
	13 or 14	Timestamp request or reply

6 of 21

پیام های ICMP به دو دسته ۱. پیام های گزارش خطا و ۲. پیام های پرس و جو (Query) تقسیم می شود.

در جدول انواع پیام ها و type و نوع پیام ها را مشاهده میکنید که با برخی از این ها آشنایی دارید.

## • قالب پیام های ICMP :



• نوع پیام : Type

• بیان دلیل این نوع پیام : Code

• کنترل خطای انتقال : Checksum

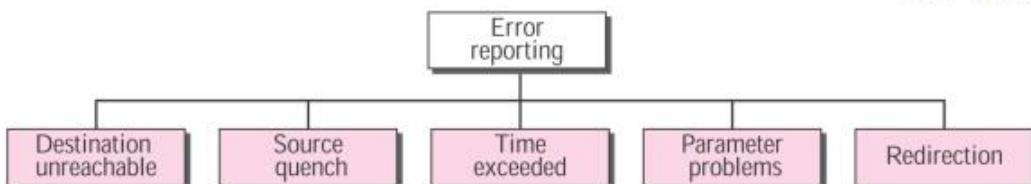
• الباقي : وابسته به نوع پیام

7 of 21

فرمت پیام ها ICMP را در این اسلاید میبینیم.

## • پیام های گزارش خطای ICMP:

### • پنج نوع خطای ICMP:



### • موارد عدم ارسال پیام گزارش خطای ICMP:

• خطای در مورد بسته حاوی پیام خطای ICMP

• قطعه های دوم به بعد یک بسته قطعه قطعه شده

• بسته های دارای آدرس چندپیخشی

• بسته های دارای آدرس های خاص مانند 127.0.0.0 با 0.0.0.0

8 of 21

در ICMP پنج نوع پیام گزارش خطای داریم(طیق نمودار) که در ادامه هر یک شرح داده می شود.

مواردی استثنای که پیام گزارش خطای ارسال نخواهد شد در تصویر ذکر شده است.

## Destination Unreachable •

- عدم توانایی مسیریاب در جلوگیری از بسته یا میزبان در تحویل بسته

Type: 3	Code: 0 to 15	Checksum
	Unused (All 0s)	
Part of the received IP datagram including IP header plus the first 8 bytes of datagram data		

- نمونه کدها :

شبکه در دسترس نیست : **Code 0** •

میزبان در دسترس نیست : **Code 1** •

9 of 21

مسیریاب نمی تواند راهی به مقصد از طریق جدول مسیریابی پیدا کند.

Type برابر ۳ هست و Code : از صفر تا ۱۵ که نمونه ای از این کد در شکل هست.

Unused : ادامه هدر را در اینجا نداریم

فیلد بعدی (Data): بخشی از IP دیتاگرام که شامل IP هدر و ۸ بیت اول دیتاگرام در این قسمت هست که مشخص میکند کدام بسته با این اتفاق مواجه شده است.

## Source Quench •

- عدم کنترل جریان و کنترل ازدحام در پروتکل IP

### اعلام ازدحام

Type: 4	Code: 0	Checksum
	Unused (All 0s)	
Part of the received IP datagram including IP header plus the first 8 bytes of datagram data		

- ارسال یک پیام Source Quench به ازای هر بسته حذف شده

- عدم وجود مکانیزم اعلام رفع ازدحام

- عدم توانایی در شناسایی مبدأ تولید کننده ازدحام در ارتباطات many-to-one

قالب بسته پیام Source Quench در تصویر گویا می باشد و توضیحات به مانند اسلاید قبل عدم وجود مکانیزم اعلام رفع ازدحام در پرتوکل IP وجود ندارد ولی در TCP وجود دارد. مبدا تولید کننده ازدحام در پیام های Source Quench قابل تشخیص نمی باشد. این خطا به دلیل سرریز بافر بوجود می آید.

## Time Exceeded •

### دو حالت:

- بخش TTL یک بسته به صفر برسد

- همه قطعات یک بسته به موقع به مقصد نرسند

Type: 11	Code: 0 or 1	Checksum
	Unused (All 0s)	
Part of the received IP datagram including IP header plus the first 8 bytes of datagram data		

بسته خطای Time Exceeded در مواردی که بسته بموقه به مقصد نرسد که یا TTL صفر شده و یا همه قطعات یک بسته بموقه به مقصد نرسد. (بر اساس مقدار code صفر یا یک این دو حالت تفکیک میشود)

## Parameter Problem •

### • حالت ها:

- خطا یا ابهامی در بخش های سرآیند. **Pointer** = بخش دارای مشکل
- عدم وجود بخش لازمی از یک گزینه **Pointer**. (option) کارایی ندارد

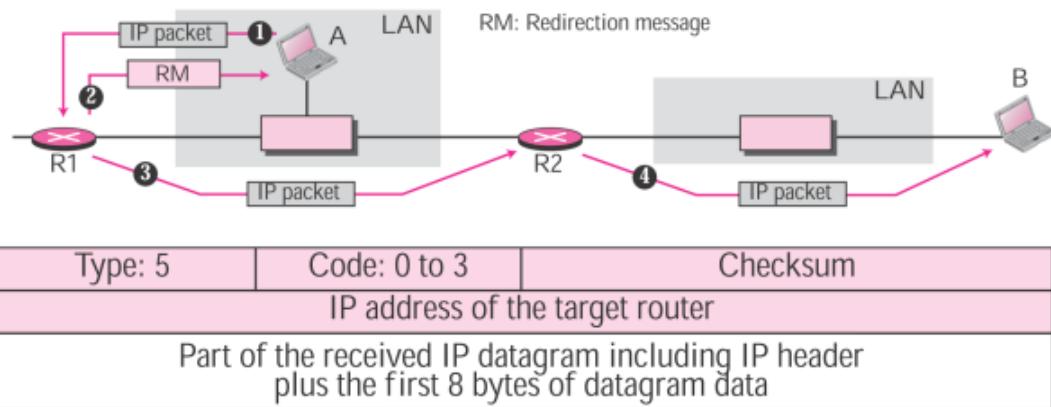
Type: 12	Code: 0 or 1	Checksum
Pointer		Unused (All 0s)
Part of the received IP datagram including IP header plus the first 8 bytes of datagram data		

در این حالت ها پیام Parameter Problem ارسال می شود.

اگر خطا یا ابهامی و جود داشته باشد فیلد Pointer به آن بخش اشاره میکند و در صورتی که بخشی از گزینه وجود نداشته باشد این فیلد هیچ کارایی ندارد.

## Redirection •

### • اصلاح مسیریاب پیش فرض برای میزبان ها



13 of 21

برای اصلاح مسیریاب پیشفرض برای میزبان ها این بسته ارسال می شود.

مثال در تصویر: فرض کنید که قبلا هاست B از طریق روتر ۱ R<sup>1</sup> قبل در دسترس بوده الان تغییر کرده مسیر و الان از طریق روتر ۲ R<sup>2</sup> امکان مسیر یابی دارد. در چنین حالتی اگر بسته ای به مقصد B به روتر ۱ R<sup>1</sup> ارسال شود روتر ۱ R<sup>1</sup> بسته را روت میکند از طریق R<sup>2</sup> ولی یک RM: بفرستد جهت اصلاح مسیر بسته ها را به R<sup>2</sup> نیز به هاست A ارسال می کند که مشخص میکند از این به بعد

اسلاید ۱۴ :

: ICMP کاربردهای

## پیام های پرس و جو (Query)

رفع برخی عیوب شبکه اتفاق می افتد. مثال: redirection اسلاید قبل

### Echo request و Echo reply

هاست ها پیام هایی هستند که در sync در هاست ها بین روتراها و

هاست ها پیام هایی هستند که در ICMP رد و بدل می شود و تعریف شده است.

در ادامه هریک شرح داده می شود.

### • پیام های پرس و جو (query)

### Echo reply و Echo request •

#### • هدف : رفع عیب

• بررسی عملکرد لایه IP

• تعیین حضور یک میزبان یا مسیریاب مشخص

Type 8: Echo request  
Type 0: Echo reply

Type: 8 or 0	Code: 0	Checksum
Identifier	Sequence number	
Optional data Sent by the request message; repeated by the reply message		

هدف از پیام های Echo request و Echo reply این هست که مسیریاب بخواهد بداند که آیا یک ماشین خاص در شبکه قابل دسترس و موجود است یا خیر. در پاسخ به دریافت echo، مقصد با ارسال پیام echo reply به آن پاسخ می دهد. با این پرسش و پاسخ، یک ماشین می تواند از قابل دسترس بودن یک مسیریاب یا ماشین میزبان در شبکه مطلع شود. یک مسیریاب یا ماشین میزبان در شبکه مطلع شود.

## • پیام های پرس و جو (query) Timestamp reply و Timestamp request •

• تعیین زمان رفت و برگشت پیام

• همزمان سازی دو گره

Type 13: request  
Type 14: reply

Type: 13 or 14	Code: 0	Checksum
Identifier	Sequence number	
	Original timestamp	
	Receive timestamp	
	Transmit timestamp	

**sending time = receive timestamp – original timestamp**

**receiving time = returned time – transmit timestamp**

**round-trip time = sending time + receiving time**

16 of 21

با پیام های Timestamp request و Timestamp reply دو کارکرد دارد هم می توانیم تایم یک هاست یا روتر را با یک peer host هماهنگ کنیم و با این کار می توانیم زمان رفت و برگشت پیام را هم تعیین کنیم و round-trip time را هم مشخص کنیم.

فیلدهای این پیام Original Timestamp و سه تا Seqquence number شامل Receive Timestamp و Transmit Timestamp و Receive Timestamp فیلد می توانیم round-trip time را محاسبه کنیم دهیم.

Sending time : زمان ارسال از فرستنده تا گیرنده

Receive Time : زمان ارسال از گیرنده به فرستنده (مسیر برگشت)

## • مثال:

original timestamp: 46

receive timestamp: 59

transmit timestamp: 60

return time: 67

sending time =  $59 - 46 = 13$  milliseconds

receiving time =  $67 - 60 = 7$  milliseconds

round-trip time =  $13 + 7 = 20$  milliseconds

Time difference = receive timestamp – (original timestamp field + one-way time duration)

Time difference =  $59 - (46 + 10) = 3$

مثال را خودتون ببینید که چجوری محاسبه شده است.

## ICMP

### • Checksum

#### • کل پیام

#### • محاسبه

• مقدار دهی اولیه به صفر

• محاسبه مجموع همه ۱۶ بیتی های سرآیند و داده

• مکمل شدن حاصل جمع

• قرار دادن در بخش **checksum**

#### • بررسی

• محاسبه مجموع همه ۱۶ بیتی ها

• مکمل شدن حاصل جمع

• اگر نتیجه صفر شد خطأ نداریم

8	0	0
1		9
TEST		
8 & 0	→ 00001000	00000000
0	→ 00000000	00000000
1	→ 00000000	00000001
9	→ 00000000	00001001
T & E	→ 01010100	01000101
S & T	→ 01010011	01010100
Sum	→ 10101111	10100011
Checksum	→ 01010000	01011100

18 of 21

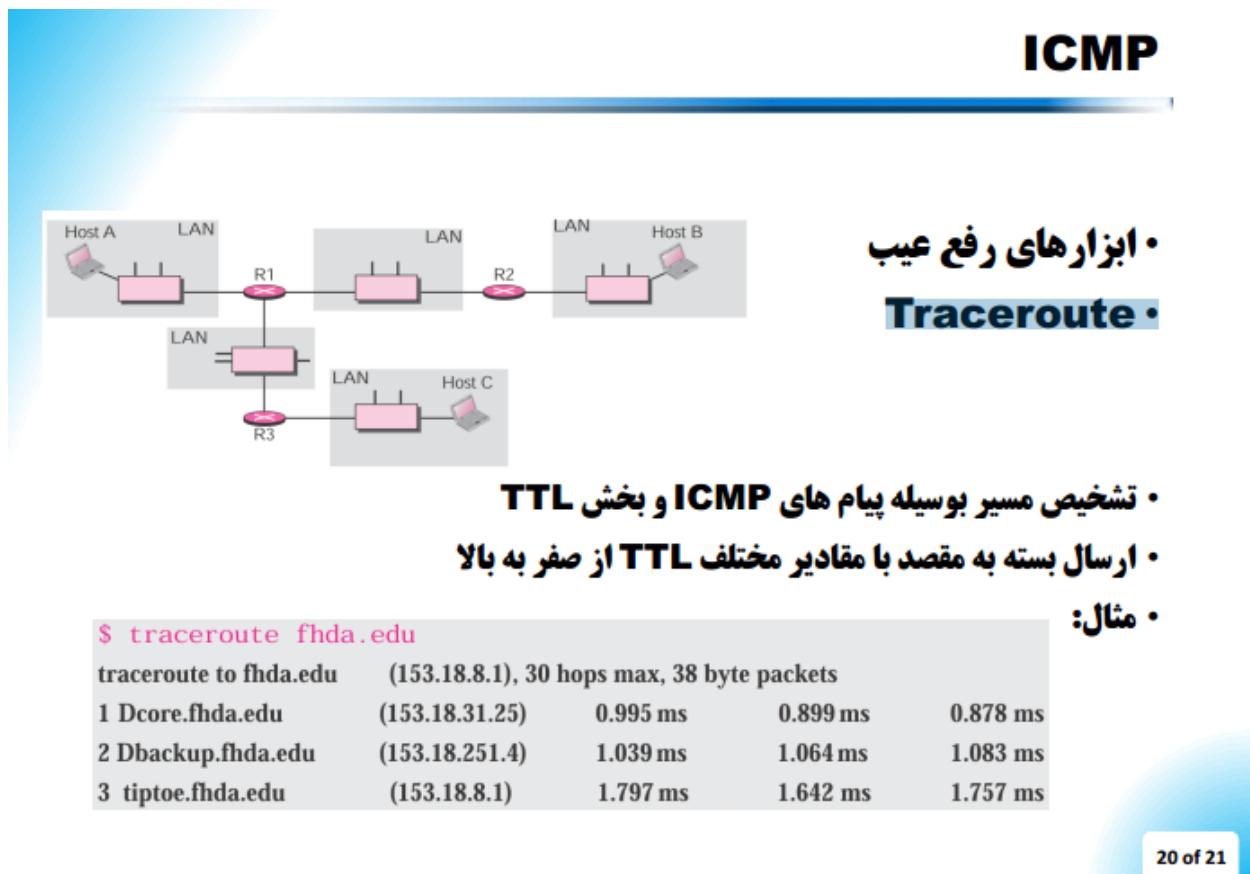
روش محاسبه فیلد Checksum هست که واضح هست و نیازی به توضیح نداره

ابزارهای رفع عیب

## ping •

نمونه ای از ping و Echo reply که یکی از روش های رفع عیب هست.

Ping به ما می گوید مقصد(Destination) وجود دارد یا خیر و زمان رفت و برگشت چقدر است.



## Traceroute •

یکی از ابزارهای رفع عیب Traceroute هست که برای تشخیص مسیر از یک مبدأ به یک مقصد هست و تعداد گام های یک مسیر را تشخیص بدید.

به دلیل قطعی زیاد ارتباط استاد و قطعی صدا استاد فایل شماره ۱۲ را در سایت جهت مطالعه قرار میدهند و تدریس نکردند و از آن قسمت سوال در امتحان پایانی نخواهد آمد.

موفق و موید باشید. ❤