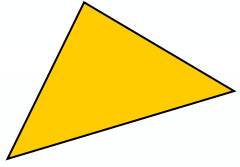


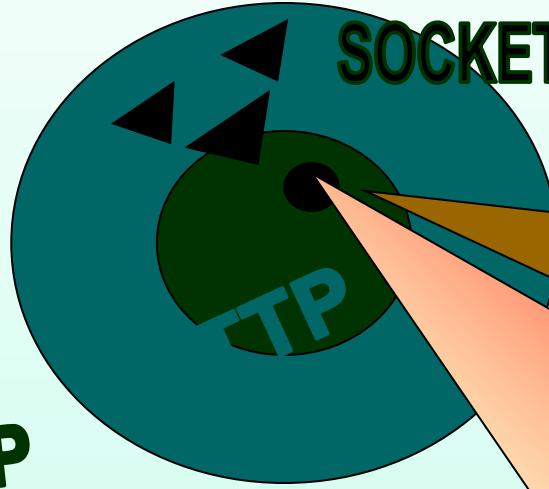
اصول مهندسی

اینترنت



HTTP

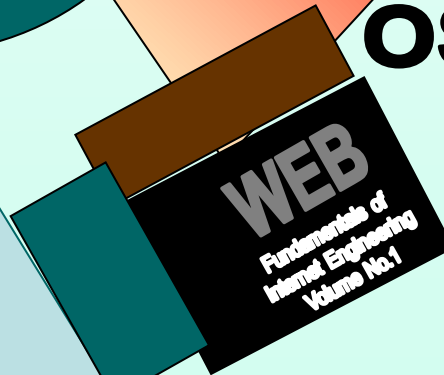
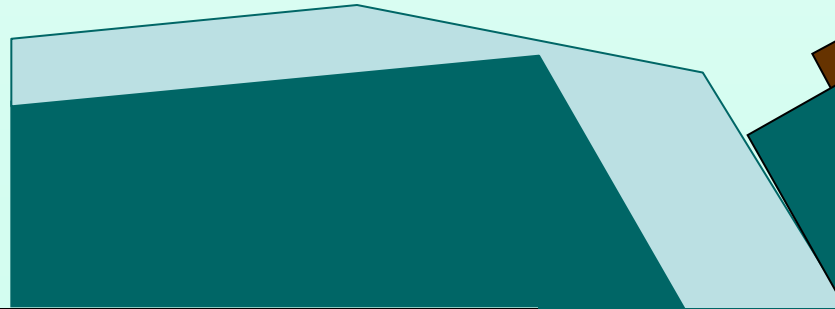
SOCKET PROGRAMMING



TCP

TCP/IP

OSPF



WEB

Fundamentals of
Internet Engineering
Volume No.1

HTTP



اصول مهندسی اینترنت

گردآوری و تالیف: مهندس احسان ملکیان



شناسنامه درس

اصول مهندسی اینترنت	نام درس:
مهندس احسان ملکیان	نام مؤلف:
هنگامه رضایی - دکتر شکيبا ضيایی	ویراستاران:
نص	انتشارات:
۳	تعداد واحد:
فصل ۱ الی ۷	فصلهای مرجع درس:
مهندسی کامپیوتر (نرم افزار)	رشته تحصیلی:
کامپیوتر	گروه آموزشی:
دکتر داود کریمزادگان مقدم	طراح اسلایدهای خلاصه درس:

جایگاه درس در رشته کامپیوتر

دروس پیش نیاز مهندسی اینترنت: معماری کامپیوتر – سیستمهای عامل

اختیاری

۳۰

۱۰

نوع درس:

تعداد کل ساعات تدریس:

تعداد جلسات تدریس:

رئوس مطالب یادگیری

مفاهیم شبکه‌های کامپیوتری

• کاربردهای شبکه‌های کامپیوتری

• سخت‌افزار شبکه

• دسته‌بندی شبکه‌ها

• روشهای برقراری ارتباط دو ماشین در شبکه

• مدل هفت‌لایه‌ای OSI

• مدل چهارلایه‌ای TCP/ IP

فصل اول: مفاهیم شبکه‌های کامپیوتری

اهداف آموزشی:

- مفهوم شبکه و کاربردهای آن
- سخت‌افزار شبکه
- انواع سوئیچینگ
- طراحی شبکه و اصول لایه‌بندی
- مدل هفت‌لایه‌ای OSI از سازمان استاندارد جهانی
- مدل چهارلایه‌ای TCP/IP



شبکه‌های کامپیوتری مجموعه‌ای از کامپیوترهای **مستقل** است که به نحوی با یکدیگر اطلاعات و داده **مبادله** می‌نمایند.

تبادل داده

ردوبدل نمودن داده بدون توجه به نوع کانال انتقال

استقلال کامپیوترها

کارکردن هر ماشین به تنهایی در صورت نبودن در شبکه

کاربردهای شبکه‌های کامپیوتری

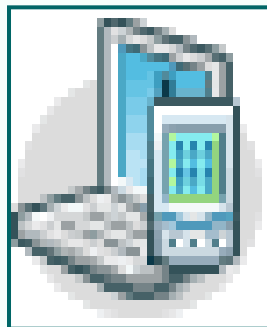
✓ اشتراک منابع

✓ حذف محدودیت‌های جغرافیایی در تبادل داده‌ها

✓ کاهش هزینه‌ها

✓ بالا رفتن قابلیت اعتماد سیستمها

✓ افزایش کارایی سیستم



خدمات معمول در شبکه

دسترسی به بانکهای اطلاعاتی راه دور

پست الکترونیکی

خدمات انتقال فایل

ورود به سیستم از راه دور

گروههای خبری

جستجوی اطلاعات مورد نیاز

تبلیغات

تجارت الکترونیکی

بانکداری الکترونیکی

سرگرمی و معاوره

مجلات و روزنامه‌های الکترونیکی

معاوره مستقیم و چهره به چهره از راه دور



کنفرانس از راه دور

یافتن اشخاص مورد نظر در جهان

تلفن ودورنگار از طریق شبکه

رادیو از طریق شبکه

آموزش از راه دور

ارائه مدون اطلاعات فنی و علمی

اخبار مربوط به هنر ، ورزش ، سیاست ، تجارت و...

کاریابی و اشتغال

درمان از راه دور

خرید و فروش روزمره با استفاده از کارت اعتباری

انجمن های خیریه

مشاوره از راه دور

دسته بندی سخت افزار شبکه های کامپیوتری



از دیدگاه
مقیاس
بزرگی

- ۱- شبکه های LAN
- ۲- شبکه های MAN
- ۳- شبکه های WAN



از دیدگاه
تکنولوژی
انتقال

شبکه های
نقطه به نقطه

شبکه های
پخش فراگیر

معایب شبکه‌های پخش فراگیر

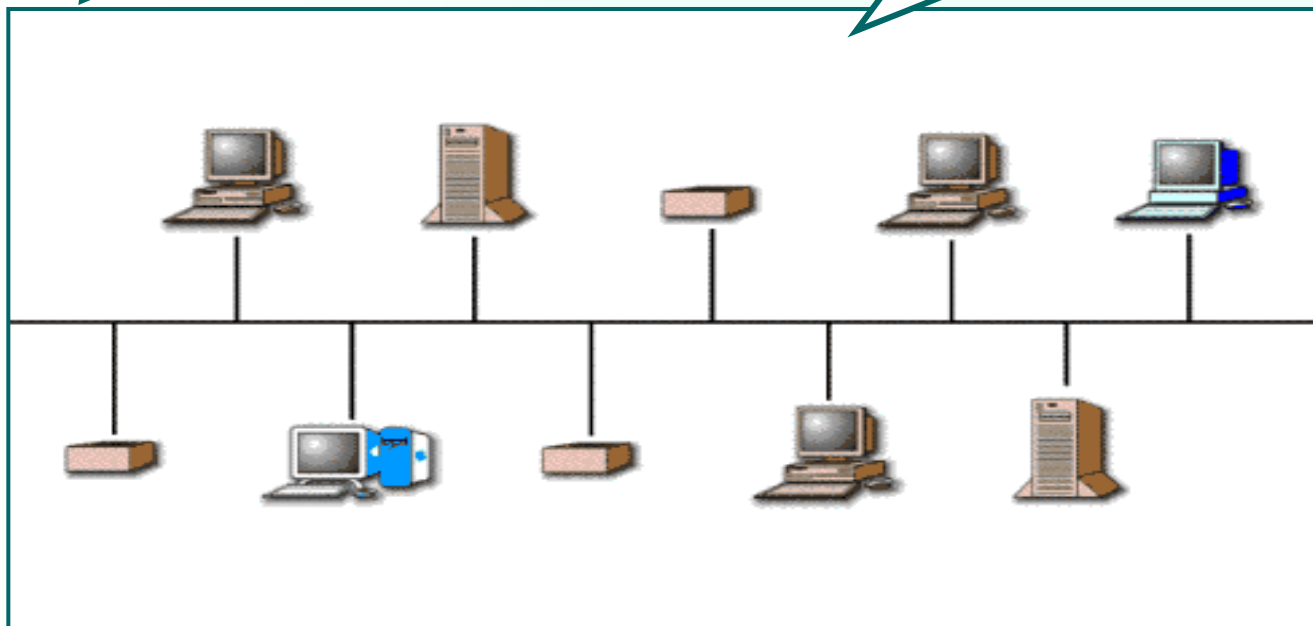
۱- مدیریت پیچیده کانال

۲- امنیت کم

۳- کارایی پایین

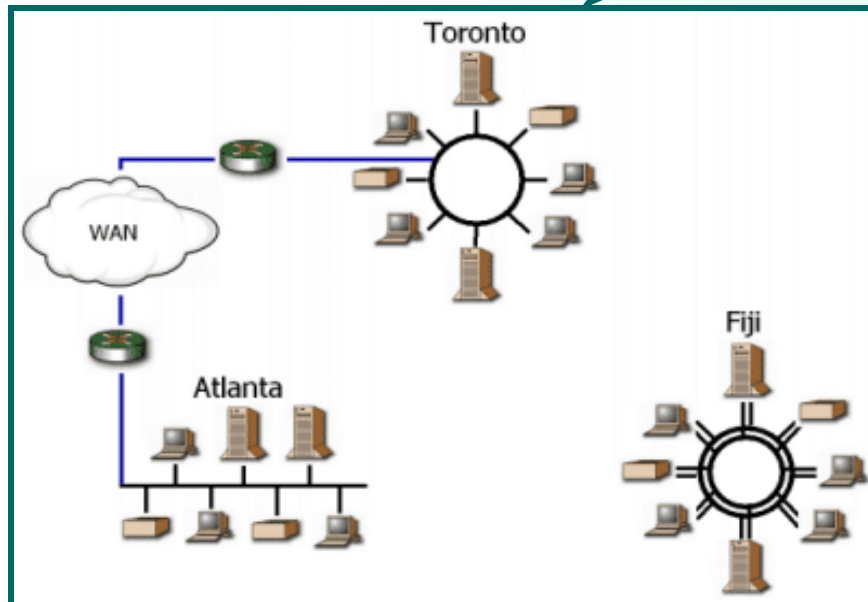
شبکه پخش فراگیر (Broadcast)

انتقال اطلاعات از طریق یک کانال
فیزیکی مشترک توسط تمام ایستگاهها



شبکه‌های نقطه به نقطه (point to point)

وجود فقط و فقط یک کانال فیزیکی و مستقیم
بین دو ماشین در شبکه



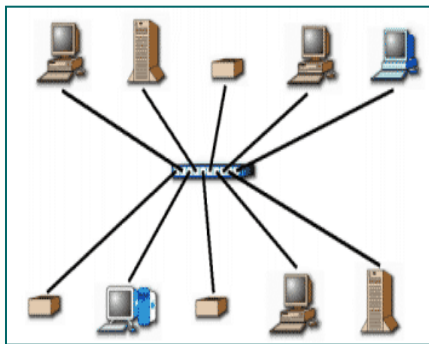
شبکه محلی LAN

- ۱- فواصل جغرافیایی محدود (حداکثر تا چند کیلومتر)
- ۲- تعداد ایستگاهها کم
- ۳- کوتاه بودن طول کانال انتقال

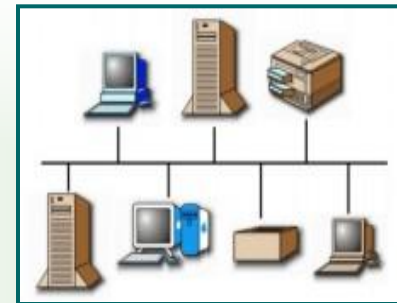


محاسن شبکه‌های LAN

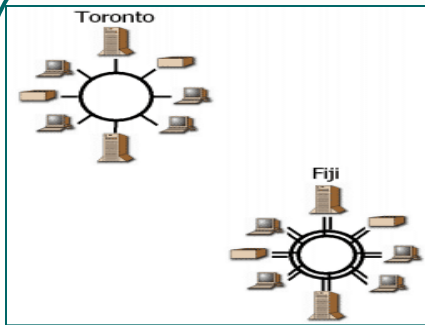
۱. افت سیگنال کم، نرخ خطای پایین، **نرخ ارسال بالا** و تأخیر انتشار بسیار ناچیز به دلیل کوتاه بودن طول کانال
۲. **مدیریت آسانتر** شبکه به علت محدود بودن تعداد ایستگاهها
۳. **هزینه پایین** نصب و راه‌اندازی این نوع شبکه.



STAR



BUS



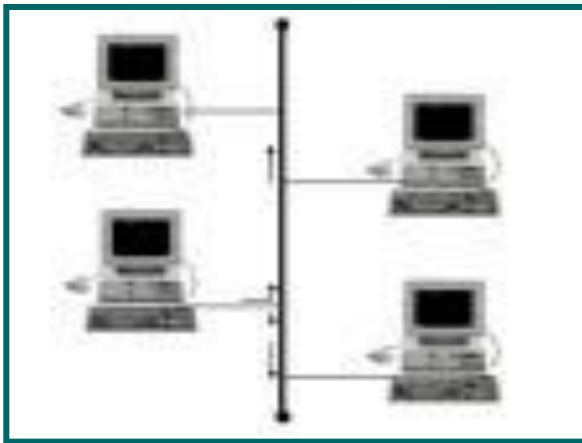
RING



انواع شبکه‌های محلی

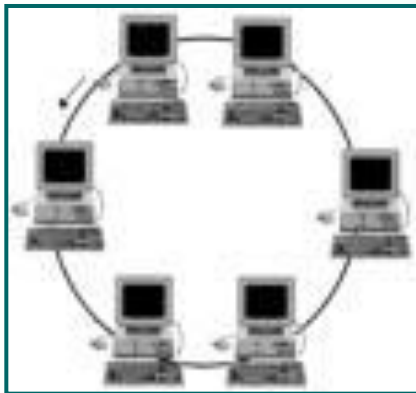
😊 اتصال تمام ایستگاهها از طریق یک کانال
فیزیکی مشترک

😊 سادگی در نصب و راه اندازی و ارزان بودن



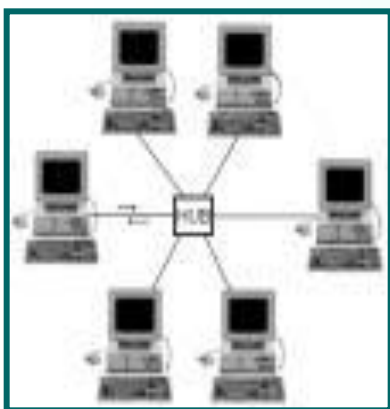
توپولوژی حلقه – (Ring)

- ☺ اتصال ایستگاهها در یک ساختار حلقوی به یکدیگر
- ☺ یکطرفه بودن ارتباط هر ایستگاه با ایستگاه بعدی خود
- ☺ دریافت بسته های اطلاعاتی توسط تمام ایستگاههای بین مسیر دو ایستگاه غیر مجاور جهت انتقال اطلاعات بین آن دو ایستگاه



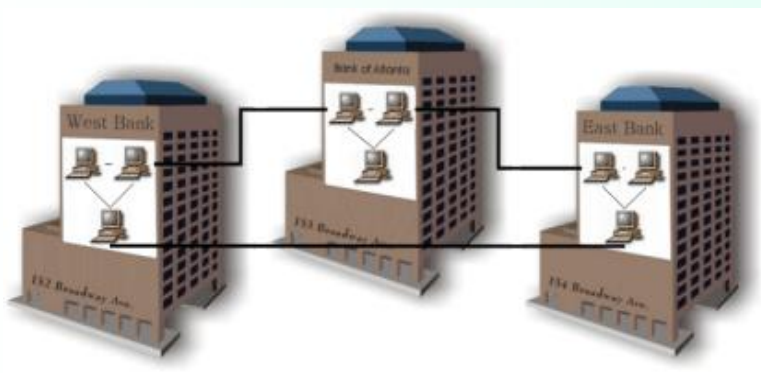
توپولوژی ستاره - (Star)

- ☺ اتصال تمام ماشینهای شبکه توسط یک گره مرکزی
- ☺ گره مرکزی میتواند سوئیچ سریع یا هاب (Hub) ویا کامپیوتر باشد.



شبکه های بین شهری (MAN)

برای ایجاد شبکه در سطح یک منطقه وسیع در حد یک شهر یا اتصال چندین شبکه محلی ، از شبکه MAN استفاده می شود . این شبکه تکنولوژی و توپولوژی مشابه با شبکه های محلی دارد. بدلیل طول زیاد کانال معمولا از فیبر نوری استفاده می شود.

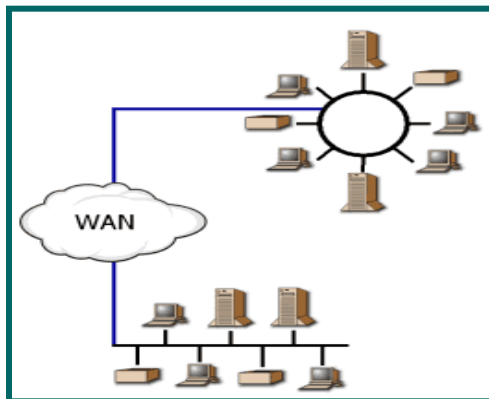


شبکه‌های گسترده (WAN)

- 😊 پیاده سازی در گستره جغرافیایی یک کشور یا جهان
- 😊 اتصال شبکه های محلی و بین شهری
- 😊 ساختار ناهمگون



توپولوژیهای مختلف شبکه های محلی
تنوع در سخت افزار و نرم افزار ماشینهای موجود
در این شبکه ها



دو بخش زیر ساخت ارتباطی در شبکه WAN

عناصر سویچ

مسیریابها: کامپیوترهای ویژه ای که پس از دریافت بسته، با در نظر گرفتن مقصد آن، کانال خروجی مناسب برای انتقال بسته به مقصد را انتخاب می نمایند.

خطوط ارتباطی یا کانالها

☺ خطوط انتقال با پهنای باند بالا
☺ برقرار کننده ارتباط عناصر

سویچ

شبکه های بی سیم (Wireless)

موارد استفاده:

- ☺ ایجاد شبکه‌ای با وجود ایستگاههای متحرک
- ☺ استفاده در مکانهایی که کابل کشی در آن مقرون به صرفه و یا عقلانی نیست.

مزایا

- ☺ ساده بودن نصب و راه اندازی این نوع شبکه

معایب

- ☺ نرخ ارسال و دریافت پایین
- ☺ نرخ خطا نسبتاً بالا
- ☺ امنیت اطلاعات کم

روشهای برقراری ارتباط دو ماشین در شبکه

۲- سوئیچینگ پیام

Message Switching

۱- سوئیچینگ مداری

Circuit Switching

۳- سوئیچینگ بسته و سلول

Packet Switching / Cell Switching

۱- سوئیچینگ مداری

Circuit Switching

لزوم برقراری اتصال فیزیکی بین مبدأ و مقصد جهت انتقال اطلاعات

معایب

- ☹ نیاز به زمان قابل توجهی برای برقراری ارتباط بین فرستنده و گیرنده
- ☹ عدم امکان برقراری ارتباط توسط ماشینهای دیگر با دو ماشین فرستنده و گیرنده هنگام اشغال بودن کانال توسط دو ماشین

۲- سوئیچینگ پیام

Message Switching

- ☺ مختص انتقال دادهای دیجیتال
- ☺ اتصال دائمی هرایستگاه با مرکز سوئیچ خود
- ☺ اضافه نمودن اطلاعات لازم به داده ها قبل از ارسال آن به مرکز سوئیچ توسط ایستگاه فرستنده
- ☺ دریافت کامل پیام توسط هر مرکز سوئیچ و انتخاب کانال خروجی مناسب بر اساس آدرس گیرنده موجود در داده

مشکل سوئیچینگ پیام

عدم محدودیت طول پیام

- ☺ بالا بودن حافظه‌های موجود در هر مرکز سوئیچ
- ☺ ارسال مجدد داده‌ها در صورت خرابی یک بیت در پیام
- ☺ تأخیر زیاد در رسیدن پیام

مزایا

- ☺ بسیار سریع و کارآمد
- ☺ عدم اشغال کانال

۳- سوئیچینگ بسته و سلول

Packet / Cell Switching

شکستن پیام توسط ایستگاه فرستنده به قطعات کوچکتری به نام **بسته** و ارسال هر بسته به همراه اطلاعات لازم برای بازسازی آن به طور جداگانه به مراکز سوئیچ

مقایسه دو روش سوئیچینگ پیام بسته / سلول

☺ مجموع تأخیر کمتر در روش سوئیچینگ بسته نسبت به روش سوئیچینگ پیام
☺ نیاز به فضای حافظه کمتر و قابل تأمین در هر مرکز سوئیچ در روش سوئیچینگ

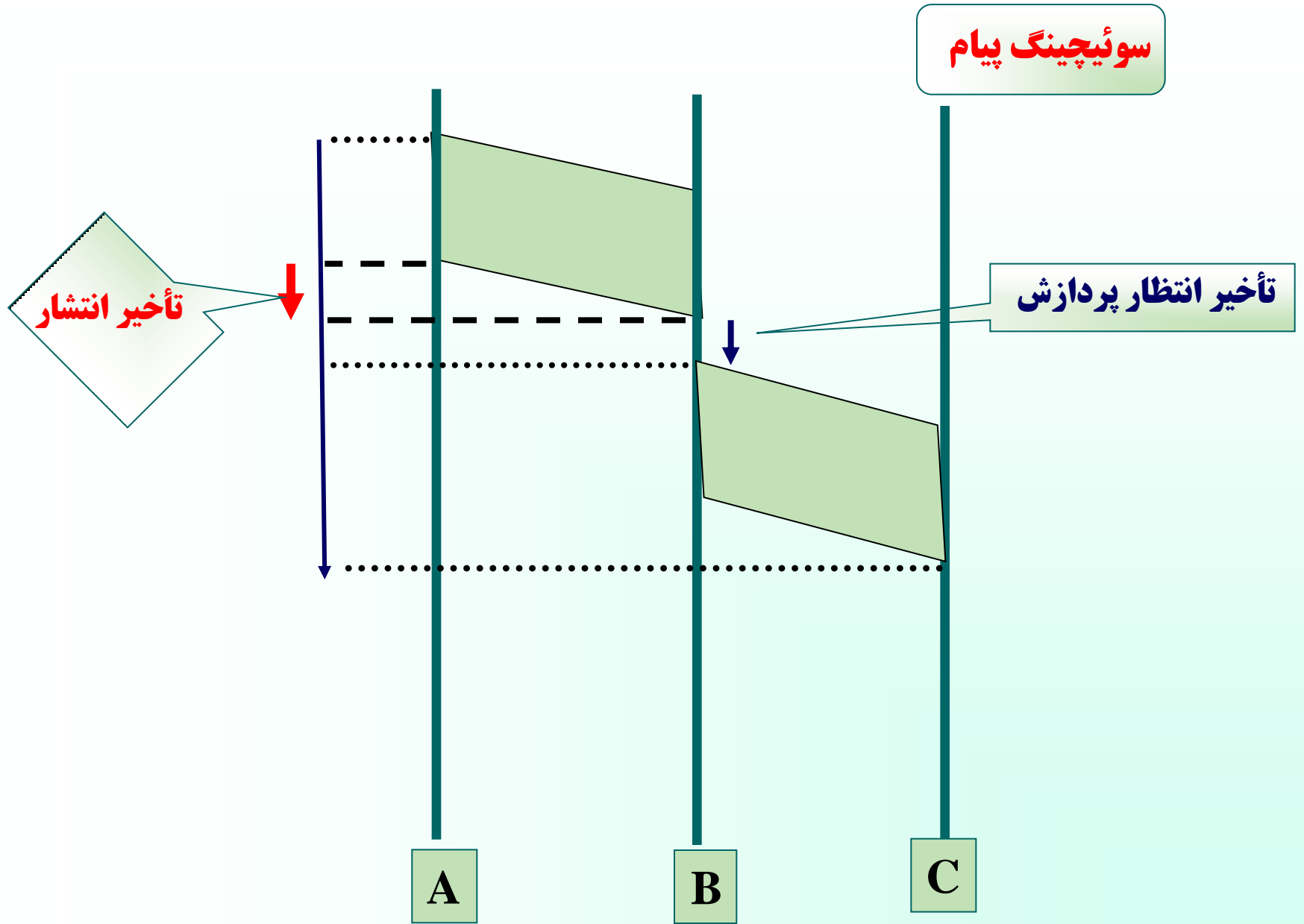
بسته

☺ عدم تأثیر خرابی یک بسته در کل پیام ارسالی و نیاز به ارسال مجدد فقط همان بسته

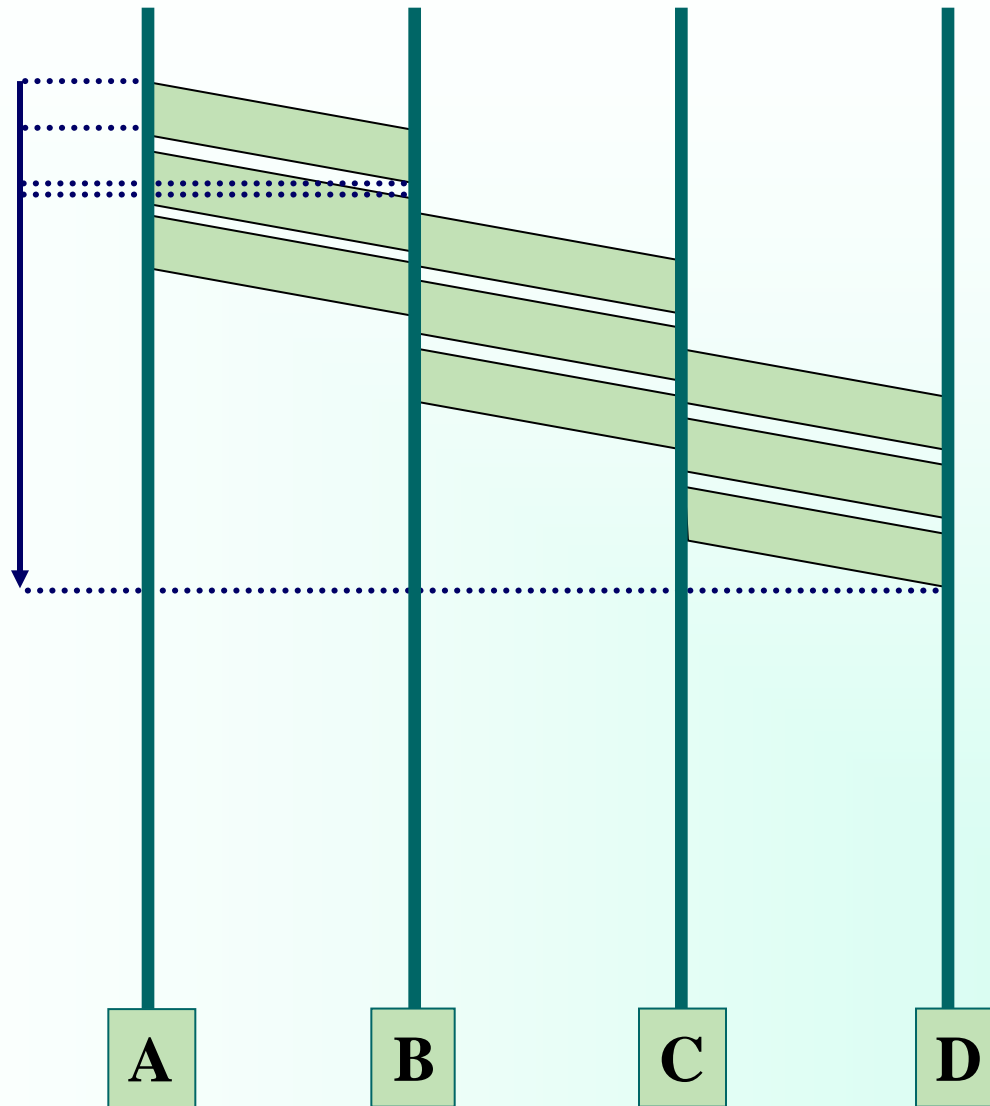
سوئیچینگ پیام

تأخیر انتظار پردازش

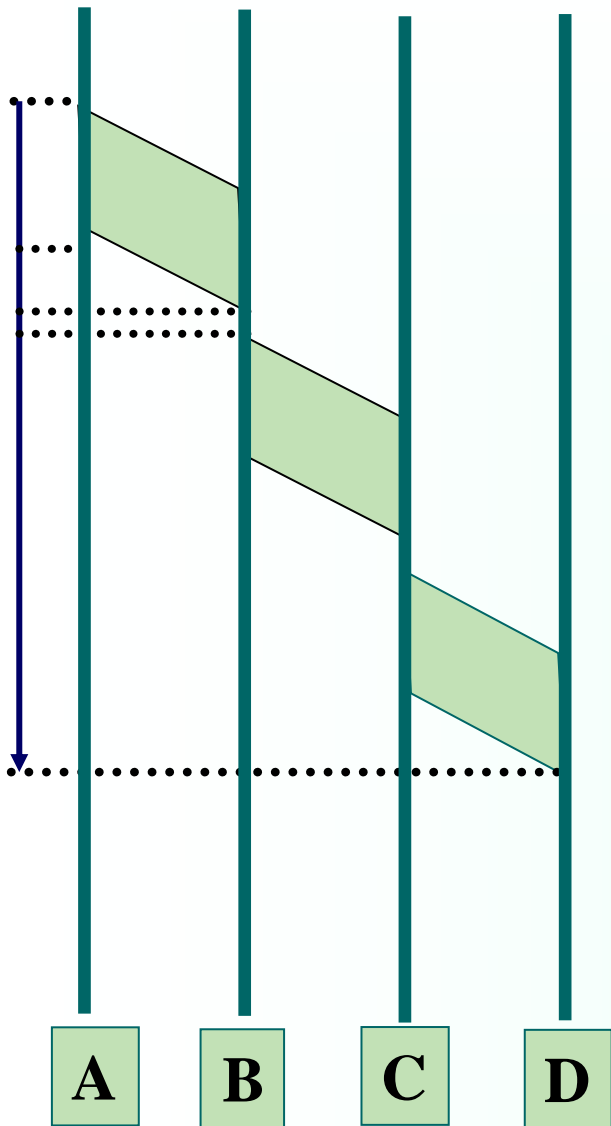
تأخیر انتشار



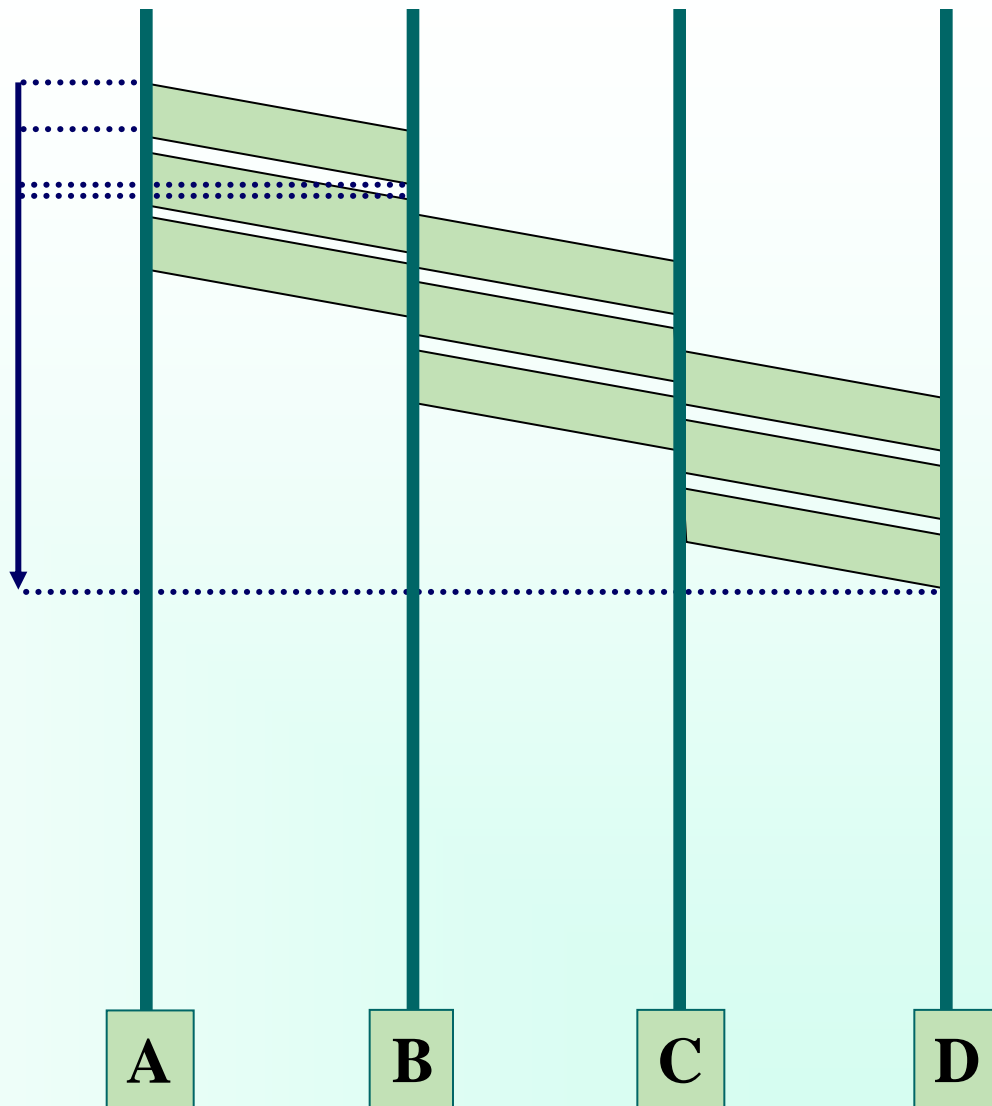
سوئچینگ بسته



سوئیچینگ پیام



سوئیچینگ بسته



زمانبندی تأخیر در روشهای سوئیچینگ پیام و بسته

طراحی شبکه ها و اصول لایه بندی

برخی از مسائل قابل توجه در طراحی شبکه ها

- ☺ چگونگی ارسال و دریافت بیت های اطلاعات
(تبدیل بیتها به یک سیگنال متناسب با کانال انتقال)
- ☺ ماهیت انتقال
- ☺ خطا و وجود نویز در کانالهای ارتباطی
- ☺ پیدا کردن بهترین مسیر و هدایت بسته ها
- ☺ تقسیم یک پیام بزرگ به واحدهای کوچکتر و بازسازی پیام
- ☺ طراحی مکانیزمهای حفظ هماهنگی بین مبدأ و مقصد
- ☺ ازدحام ، تداخل و تصادم در شبکه ها

انواع ارتباط میان دو ایستگاه

☺ ارتباط یکطرفه - Simplex:

یکطرف همیشه گیرنده و یکطرف همیشه فرستنده

☺ ارتباط دوطرفه غیرهمزمان - Half duplex

هر دو ماشین هم می توانند فرستنده باشند و هم گیرنده ولی نه بصورت همزمان

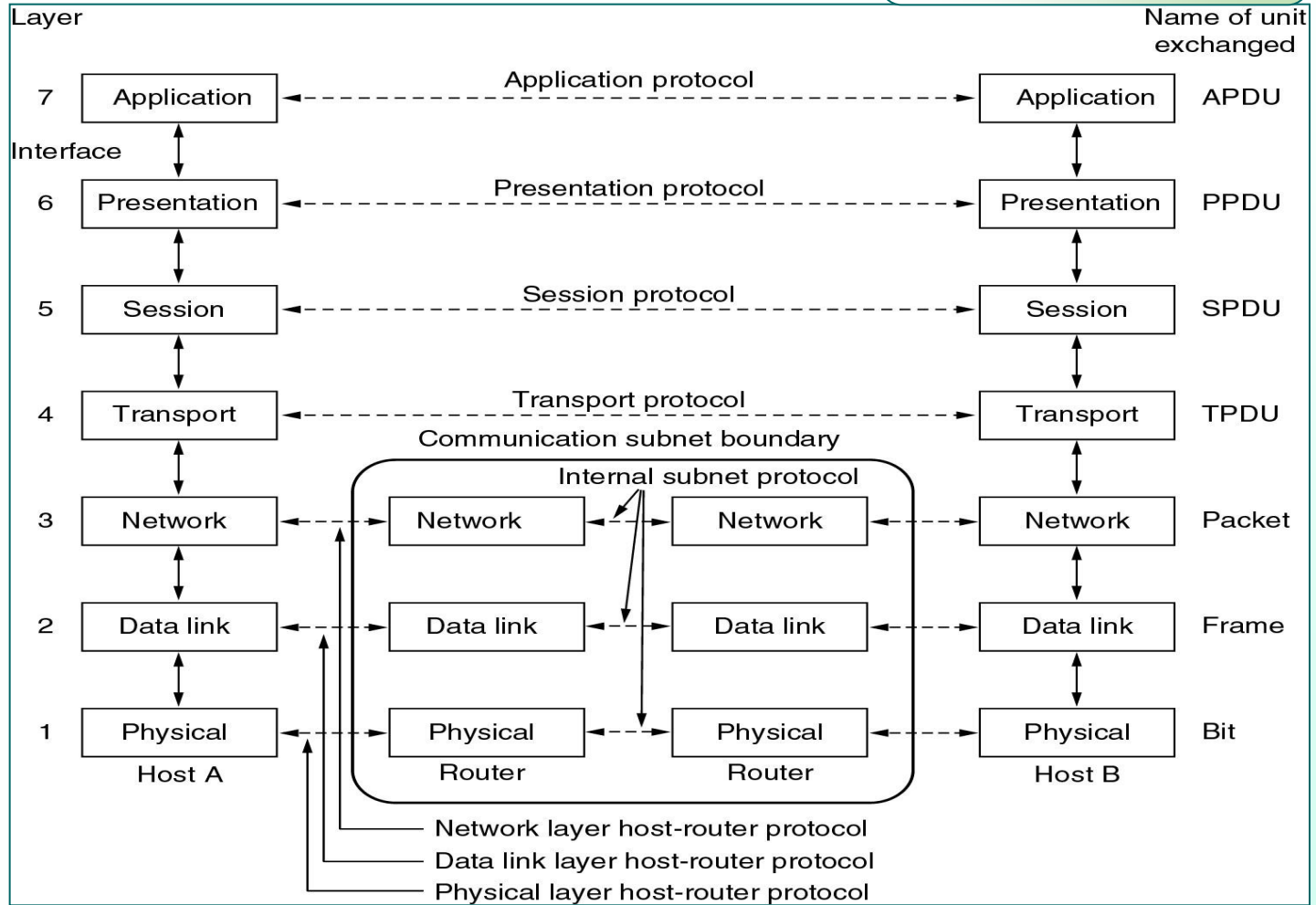
☺ ارتباط دوطرفه همزمان - Full duplex

ارتباط دو طرفه همزمان مانند خطوط ماکروویو

مدل هفت لایه‌ای OSI از سازمان استاندارد جهانی ISO

- ☺ لایه فیزیکی **Physical layer**
- ☺ لایه پیوند داده‌ها **Data link layer**
- ☺ لایه شبکه **Network layer**
- ☺ لایه انتقال **Transport layer**
- ☺ لایه جلسه **Session layer**
- ☺ لایه ارائه (نمایش) **Presentation layer**
- ☺ لایه کاربرد **Application layer**

مدل هفت لایه‌ای OSI



لایه فیزیکی Physical Layer

- ☺ انتقال بیتها به صورت سیگنال الکتریکی و ارسال آن بر روی کانال
- ☺ واحد اطلاعات : بیت

پارامترهای قابل توجه :

- ☺ ظرفیت کانال فیزیکی و نرخ ارسال
- ☺ نوع مدولاسیون
- ☺ چگونگی کوپلاژ با خط انتقال
- ☺ مسائل مکانیکی و الکتریکی مانند نوع کابل، باند فرکانسی، نوع رابط (کانکتور) کابل

لایه پیوند داده - Data Link Layer

وظایف :

- به مقصد رساندن داده‌ها روی یک کانال انتقال بدون خطا و مطمئن با استفاده از مکانیزمهای کشف و کنترل خطا.
- شکستن اطلاعات ارسالی از لایه بالاتر به واحدهای استاندارد و کوچکتر و مشخص نمودن ابتدا و انتهای آن از طریق نشانه‌های خاصی بنام **Delimiter**.
- کشف خطا از طریق اضافه کردن بیت‌های کنترل خطا
- کنترل جریان یا تنظیم جریان ارسال فریمها (مکانیزمهای هماهنگی بین مبدأ و مقصد)
- اعلام وصول یا عدم رسیدن داده‌ها به فرستنده
- وضع قراردادهائی برای جلوگیری از تصادم سیگنالهای ارسالی (این قراردادها در زیرلایه‌ای بنام **MAS** تعریف شده است)
- کنترل سخت‌افزار لایه فیزیکی

لایه شبکه

- سازماندهی اطلاعات بصورت بسته و ارسال جهت انتقال مطمئن به لایه پیوند داده‌ها
- تعیین مسیر هر بسته ارسالی برای رسیدن به مقصد
- جلوگیری از ازدحام و ترافیک در بین مسیریابها و سوئیچها
- اختصاص آدرسهای مشخص و استاندارد برای هر بسته آماده ارسال
- این لایه بدون اتصال است.

لایه انتقال

- ارسال یک بسته ویژه قبل از ارسال بسته‌ها برای اطمینان از آمادگی گیرنده برای دریافت اطلاعات
- شماره‌گذاری بسته‌های ارسالی برای جلوگیری از گم‌شدن یا ارسال دوباره بسته‌ها
- حفظ ترتیب جریان بسته‌های ارسالی
- آدرس‌دهی پروسه‌های مختلفی که روی یک ماشین واحد اجرا می‌شوند.
- تقسیم پیام‌های بزرگ به بسته‌های اطلاعاتی کوچکتر
- بازسازی بسته‌های اطلاعاتی و تشکیل یک پیام کامل
- شماره‌گذاری بسته‌های کوچکتر جهت بازسازی
- تعیین و تبیین مکانیزم نامگذاری ایستگاه‌های موجود در شبکه

لایه جلسه Session Layer

- برقراری و مدیریت یک جلسه
- شناسائی طرفین
- مشخص نمودن اعتبار پیامها
- اتمام جلسه‌ها
- حسابداری مشتریها

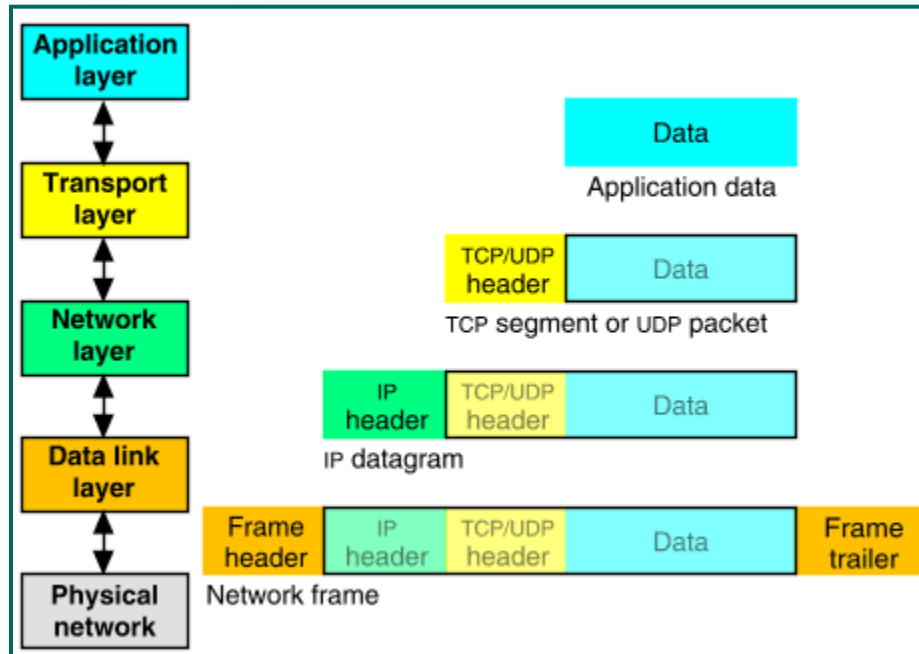
لایه ارائه (نمایش)

- فشرده‌سازی فایل
- رمزنگاری برای ارسال داده‌های محرمانه
- رمزگشائی
- تبدیل کدها به یکدیگر هنگام استفاده دو ماشین از استانداردهای مختلفی برای متن

لایه کاربرد Application Layer

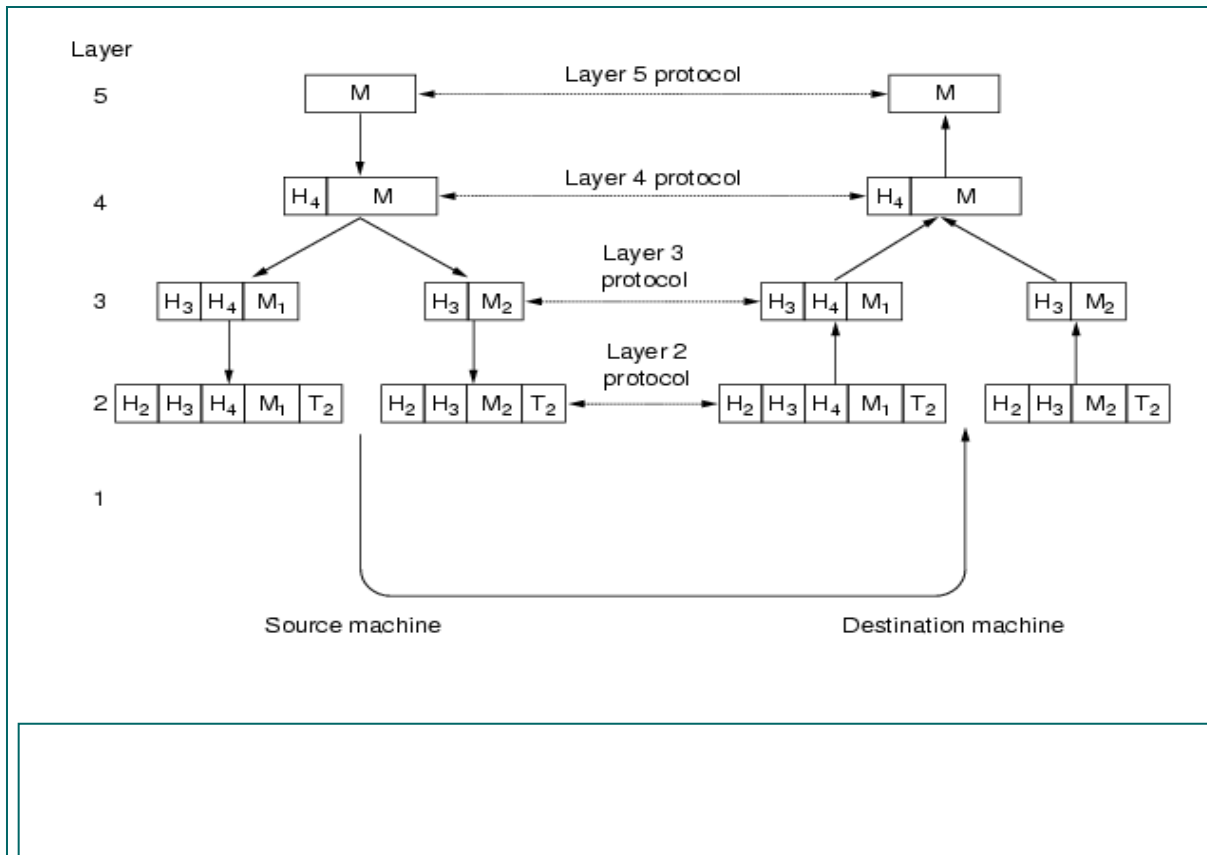
تعریف استانداردهائی نظیر :

- انتقال نامه‌های الکترونیکی
- انتقال مطمئن فایل
- دسترسی به بانکهای اطلاعاتی راه دور
- مدیریت شبکه
- انتقال صفحه وب

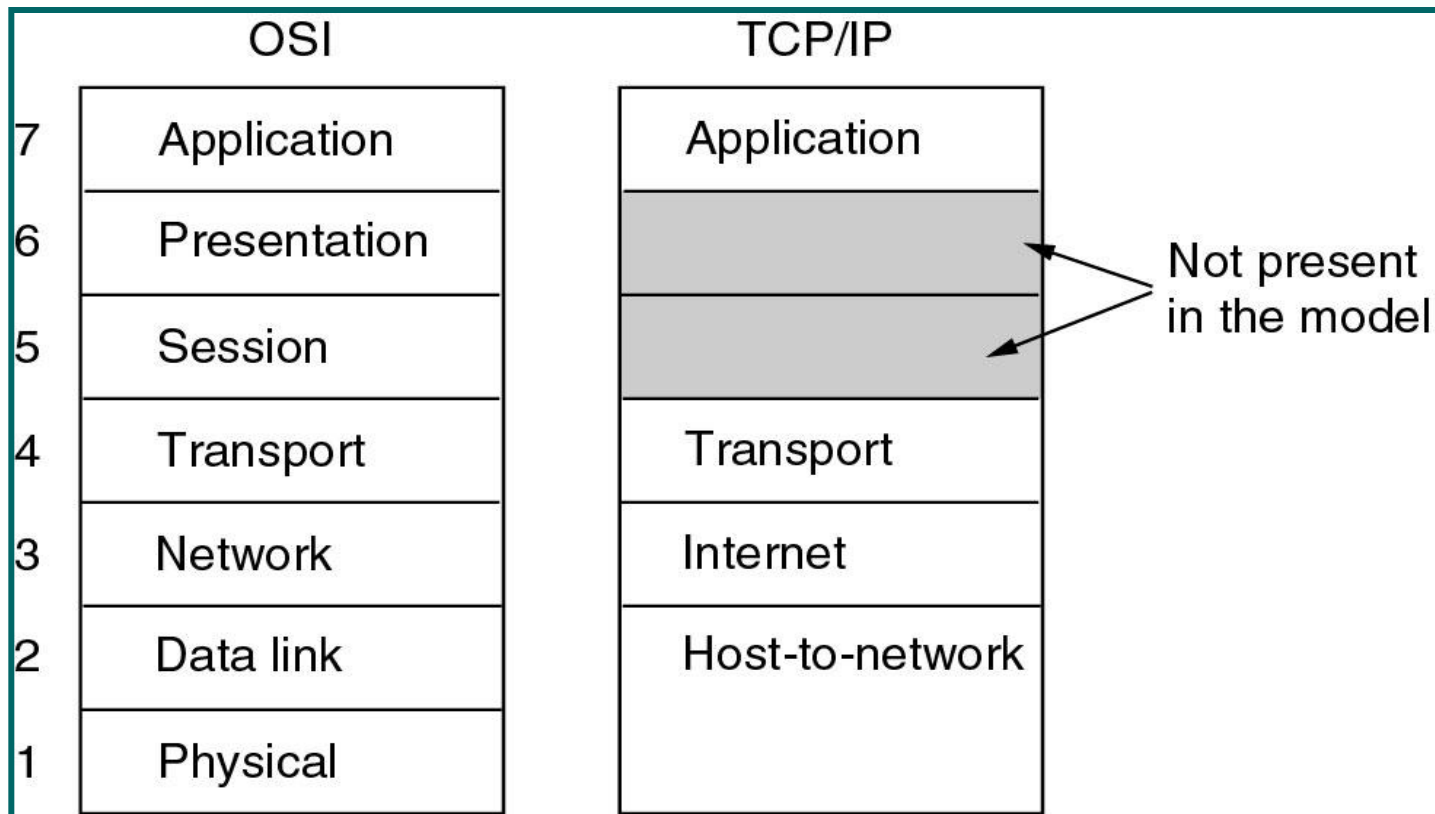


مدل OSI

روند حذف و اضافه شدن سرآیند در هر لایه



مدل چهار لایه‌ای TCP/IP



لایه‌های مدل TCP/IP

نامهای معادل در برخی از کتب	لایه‌ها
● لایه سرویسهای کاربردی	لایه کاربرد Application layer
● لایه ارتباط میزبان به میزبان (Host to Host) ● لایه ارتباط عناصر انتهائی (End to End Connection)	لایه انتقال Transport layer
● لایه اینترنت ● لایه ارتباطات اینترنت	لایه شبکه Network layer
● لایه میزبان به شبکه (Host to Network) ● لایه رابط شبکه	لایه دسترسی به شبکه Network Interface

لایه اول از مدل TCP/IP : لایه واسط شبکه

تعریف لایه‌های استاندارد سخت‌افزار، نرم‌افزارهای راه‌انداز و پروتکل‌های شبکه در این لایه. پروتکل‌هایی که در لایه اول از مدل TCP/IP تعریف می‌شوند، می‌توانند مبتنی بر ارسال رشته بیت یا مبتنی بر ارسال رشته بایت باشند.

لایه دوم از مدل TCP/IP : لایه شبکه

- بسته‌های IP بسته‌های اطلاعاتی در این لایه
- هدایت بسته‌های IP روی شبکه از مبدأ تا مقصد که این عمل از نوع بدون اتصال می‌باشد
- ویژگی ارسال چندپخشی یعنی ارسال یک یا چند بسته اطلاعاتی به چندین مقصد گوناگون در قالب یک گروه سازماندهی شده
- پروتکل‌هایی که در این لایه استفاده می‌شوند عبارتند از:
و . . . IP , IGMP , BOOTP , ARP , RARP , RIP , ICMP

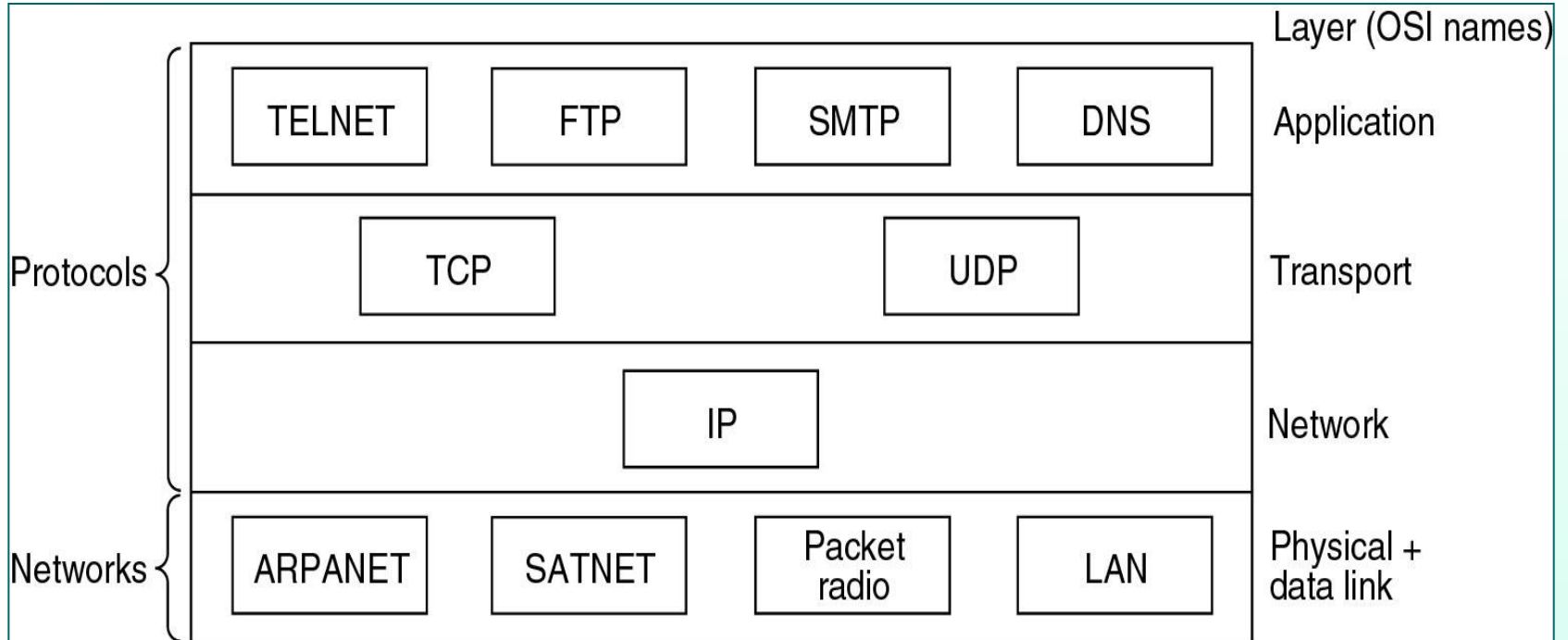
لایه سوم از مدل TCP/IP : لایه انتقال

برقراری ارتباط از طریق یک سرویس اتصال‌گرا و مطمئن با ماشینهای انتهایی یا میزبان. ارسال و یا دریافت داده‌های تحویلی به این لایه توسط برنامه‌های کاربردی و از طریق توابع سیستمی

لایه چهارم از مدل TCP/IP : لایه کاربرد

خدماتی که در این لایه صورت می‌گیرد در قالب پروتکل‌های استاندارد زیر به کاربر ارائه می‌شود :
شبیه‌سازی ترمینال
انتقال فایل یا **FTP**
مدیریت پست الکترونیکی
خدمات انتقال صفحات ابرمتنی

پروتکل‌های رایج در لایه‌ها



فصل دوم: لایه واسط شبکه

هدفهای آموزشی:

- لایه شبکه و مسائل خطوط انتقال داده
- استانداردهای انتقال روی خطوط نقطه به نقطه
 - پروتکل SLIP
 - پروتکل PPP
- استانداردهای انتقال در شبکه های با کانال مشترک
 - IEEE 802.3 CSMA/CD
 - IEEE 802.4 Token Bus
 - IEEE 802.5 Token Ring
 - IEEE 802.6 DQDB
 - IEEE 802.11 Wireless LAN

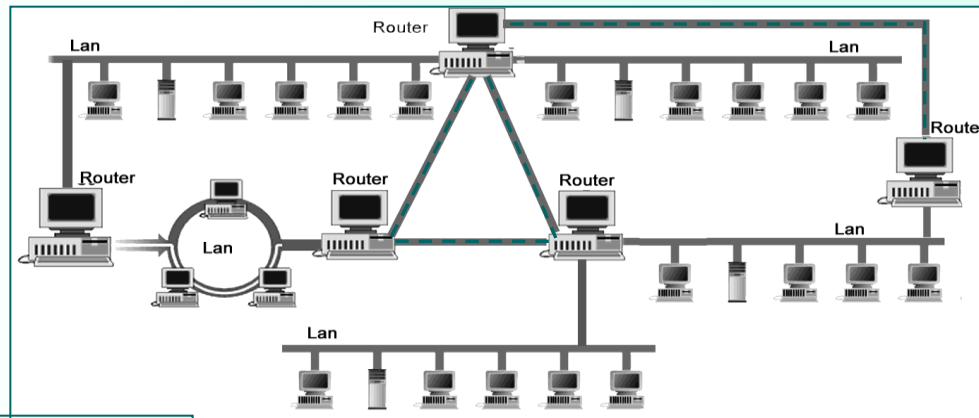


(۱) لایه واسط شبکه

- ☺ تبدیل کانال دارای خطا به یک خط مطمئن و بدون خطا
- ☺ فریم بندی اطلاعات

بسته IP

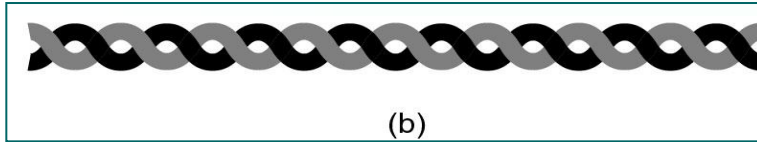
- ☺ ساختمان داده‌ای است درون فیلد داده فریمها
- ☺ عدم تغییر بسته IP با وجود تغییر شبکه و تغییرات مداوم فریم



وظیفه سخت افزار انتقال در لایه واسط شبکه: انتقال بیت‌های داده بر روی کانال فیزیکی بدون توجه به نوع و محتوای داده‌ها

کانالهای انتقال

- خطوط تلفن
- فیبرهای نوری
- سیمهای به هم بافته شده زوجی
- کابلهای هم محور (کواکسیال)
- کانالهای ماهواره‌ای
- کانالهای رادیویی
- امواج طیف نوری

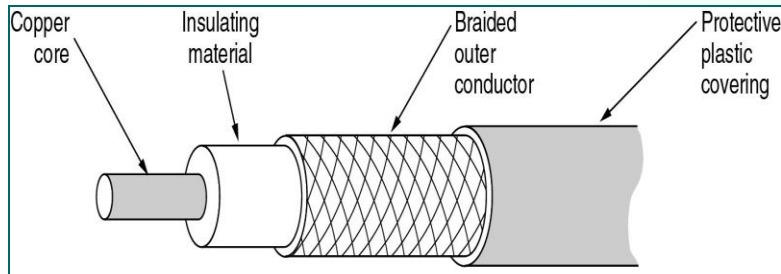


(a) Category 3 UTP.
(b) Category 5 UTP.

سیمهای به هم بافته شده زوجی:

• **UTP**: یک زوج سیم معمولی به هم بافته شده

• **STP**: یک زوج سیم معمولی به هم بافته شده
به همراه یک پوشش آلومینیمی بر روی آنها جهت
کاهش اثر نویزهای محیطی بر روی سیم



کابلهای هم محور (کواکسیال):

در انواع مختلف مانند:

Thick Coaxial Cable کابل کواکس ۵۰ اهم ضخیم

Thin Coaxial Cable کابل کواکس ۵۰ اهم نازک

کابل کواکس ۷۵ اهم معمولی)

کانالهای ماهواره‌ای : در باندهای فرکانسی مختلف
مانند:

• باند C

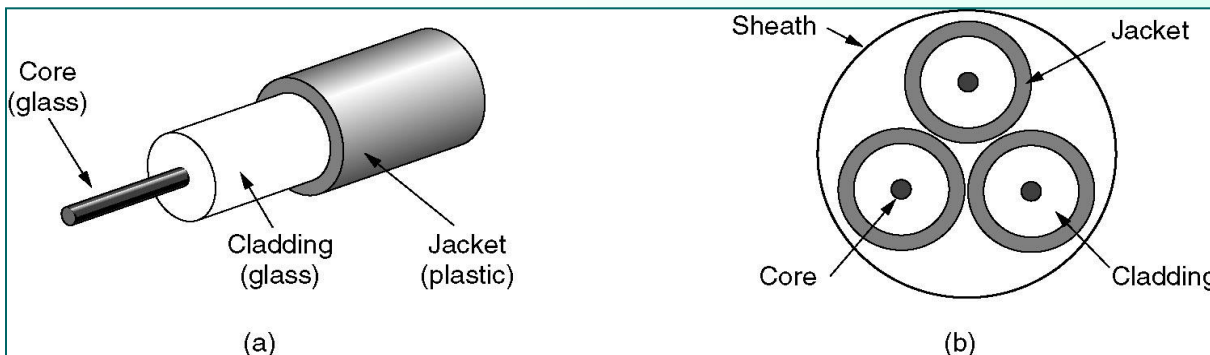
• باند Ku

• باند Ka

کانالهای رادیویی : شامل باندهای فرکانسی مختلف مثل UHF ، VHF

امواج طیف نوری : شامل نور مادون قرمز

فیبرهای نوری : در انواع مختلف مثل فیبر تک‌موده و چندموده



پهنای باند:

توانایی و ظرفیت کانال در ارسال اطلاعات با نرخ B بیت در هر ثانیه

رابطه شانون:

$$C=B.\log_2(1+S/N)$$

C : ظرفیت کانال بر حسب بیت بر ثانیه

S : متوسط توان سیگنال

N : متوسط توان نویز

B : پهنای باند کانال بر حسب هرتز

مالتی پلکسی یا تسهیم: تقسیم پهنای باند یک کانال بین چند ایستگاه

Frequency Division Multiplexing

• تسهیم در میدان فرکانس یا **FDM**

Time Division Multiplexing

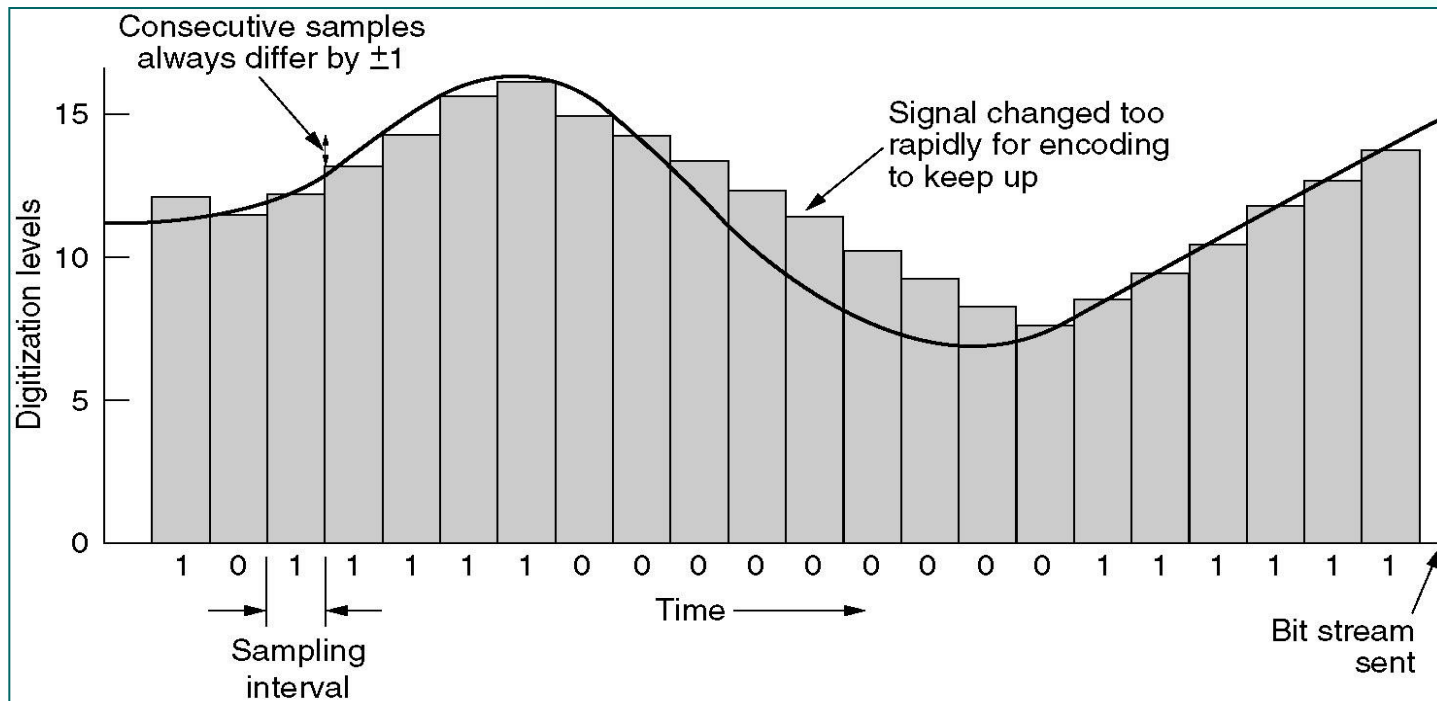
• تسهیم در میدان زمان یا **TDM**

FDM: تقسیم پهنای باند فرکانسی به N باند مجزا (N تعداد ایستگاه موجود در شبکه)

TDM: تقسیم زمان به بازه‌های کوچک (ارسال اطلاعات بر روی کانال توسط هر ایستگاه فقط در بازه زمانی مشخص)

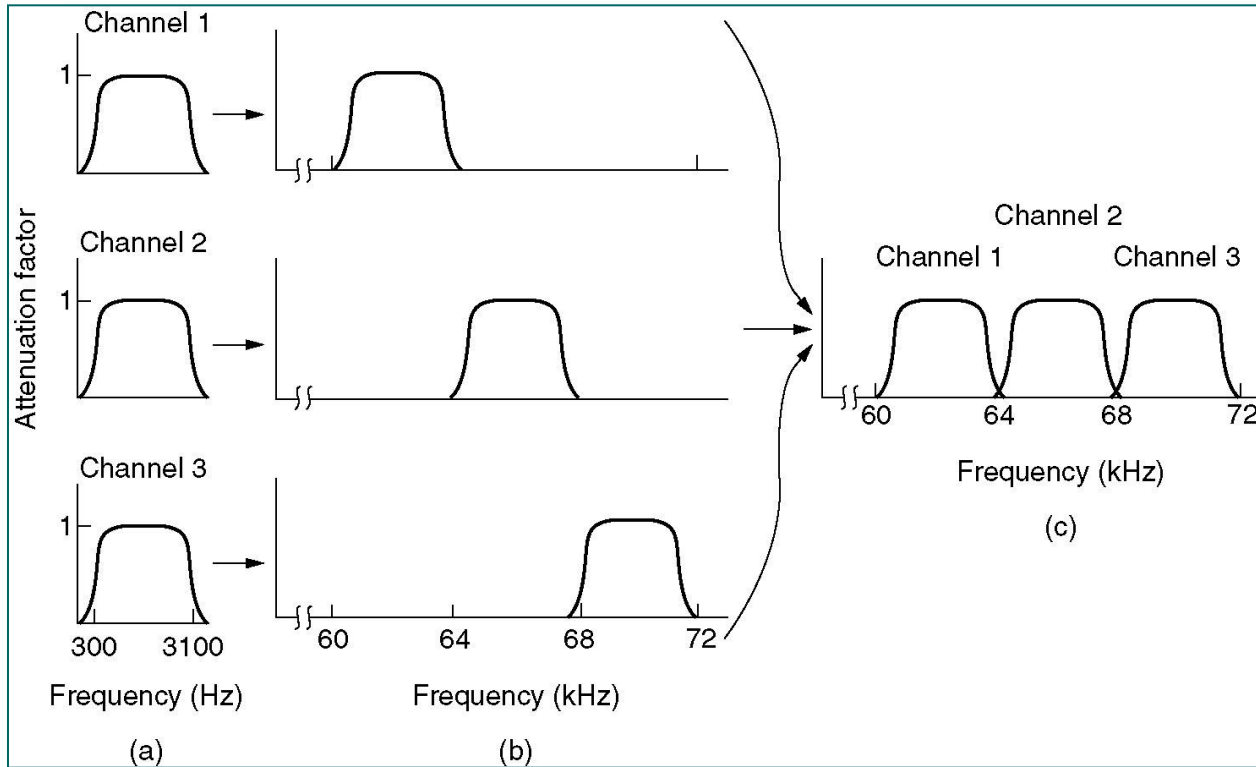
موارد کاربرد روشهای FDM و TDM :

- تعداد ایستگاهها ثابت و محدود
- ارسال حجم ثابت و دائمی داده توسط هر ایستگاه بر روی کانال



TDM

FDM



انواع خطا در شبکه‌های کامپیوتری

- نویز حرارتی
- شوک‌های الکتریکی
- نویز کیهانی

روشهای کشف خطا

- اضافه کردن بیت توازن به داده‌ها
- روش **Checksum**
- کدهای کشف خطای **CRC**

بیت توازن

- ساده‌ترین روش کشف خطا
- اضافه نمودن یک بیت توازن به ازای هر بایت از اطلاعات
- انتخاب بیت توازن به گونه‌ای که مجموع تعداد بیت‌های ۱ همیشه زوج یا فرد باشد
- این روش در صورتی موثر است که تعداد خطاهای رخ داده زوج نباشد

	01101001	بایت اصلی:
Odd Parity 1	0110100	بیت توان فرد
Even Parity 0	01101001	بیت توان زوج

روش Checksum

- جمع (XOR) تمام بایتهای یک فریم ارسالی توسط فرستنده و ایجاد بایت **Checksum**
- این روش در صورتی قادر به کشف خطا است که تعداد خطاهای رخ داده در بایتهای هم ارزش زوج نباشد

کدهای کشف خطای CRC

- محاسبه تعدادی بیت کنترلی به نام **CRC (Cyclic Redundancy Check)** به ازای مجموعه‌ای از بیتها و اضافه شدن به انتهای فریم
- مبنای کار : تقسیم چند جمله‌ای

استانداردهای انتقال روی خطوط نقطه به نقطه

- ۱۰ پروتکل Serial Line IP : SLIP
- ۲۰ پروتکل Point to Point : PPP

۱) پروتکل SLIP

روش کار:

۱. ارسال علامت مشخصه یک بایتی **0xC0** روی خط توسط ایستگاه
۲. انتقال داده بر روی خط
۳. ارسال مجدد علامت مشخصه **0xC0** جهت مشخص نمودن انتهای فریم

قالب هر فریم



معایب پروتکل SLIP

- عدم وجود کد کشف خطا در این پروتکل
- قرار گرفتن فقط بسته‌های IP درون فیلد داده فریم
- عدم پشتیبانی بسیاری از سیستم‌عاملها از این پروتکل
- لزوم داشتن آدرسهای IP ثابت و شناخته شده برای هر دو ایستگاه برقرارکننده ارتباط
- عدم تأیید و احراز هویت کاربر برقرارکننده ارتباط در این پروتکل

پروتکلی بسیار سریع به دلیل نداشتن فیلدهای سرآیند اضافی

مراحل برقراری ارتباط از طریق خط سریال نقطه به نقطه:

فاز مذاکره

Negotiation

- شماره‌گیری به کمک مودم
- اتصال تلفن توسط مودم طرف مقابل
- تبادل بسته‌های اطلاعاتی کنترلی **LCP** بین طرفین
- فریم‌های **LCP** حاوی اطلاعات پارامترهای پروتکل **PPP**
- تبادل بسته‌های **NCP** جهت تنظیم پارامترهای لایه بالاتر
- آغاز مبادله فریمها

٢) قالب فریم پروتکل PPP

Bytes	1	1	1	1 or 2	Variable	2 or 4	1
	Flag 01111110	Address 11111111	Control 00000011	Protocol	Payload	Checksum	Flag 01111110

Address Field

- مقدار فیلد تماماً ۱
- آدرس فراگیر

Control Field

- مقدار این فیلد در مورد فریمهای عادی = **00000011**
- نشان دهنده آن است که این فریم شماره گذاری شده نیست و نیازی به ارسال پیغام **ACK** توسط طرفین برای فریمها نمی باشد

Protocol

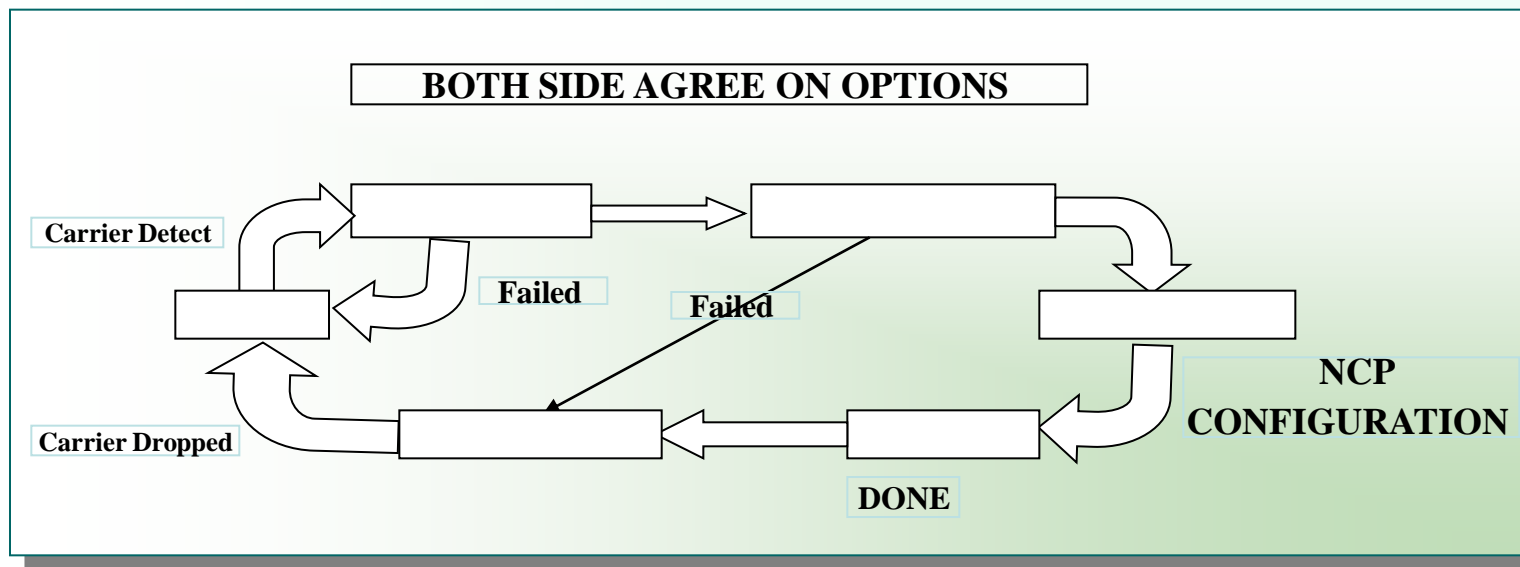
مشخص کننده آنکه بسته درون فیلد داده مربوط به چه پروتکلی در لایه بالاتر است.

Checksum

- به طور پیش فرض ۲ بایتی
- جهت کشف خطاهای احتمالی در فریم

Payload

- سایز پیش فرض این فیلد = ۱۵۰۰ بایت
- بسته مربوط به لایه بالاتر در این فیلد قرار می گیرد



PPP مراحل برقراری و ختم یک ارتباط در پروتکل

بسته های مهم LCP
Link Control Protocol

I : پیشنهاددهنده

R : پاسخ دهنده

نام بسته	جهت	عملکرد
Configure Request	I → R	لیستی از گزینه‌ها و مقادیر را برای تنظیم، پیشنهاد می‌کند.
Configure Ack	I ← R	مشخص می‌کند که تمامی پیشنهادات پذیرفته شد.
Configure Nack	I ← R	برخی از پارامترها و گزینه‌ها پذیرفته نشد.
Configure Reject	I ← R	برخی از پارامترها قابل بحث و توافق نیستند.
Terminate Request	I → R	تقاضا برای خاتمه و قطع ارتباط
Terminate Ack	I ← R	موافقت برای قطع ارتباط و کانال
Code-Reject	I ← R	تقاضایی رسیده است که شناسایی و فهم نمی‌شود.
Echo Request	I → R	لطفاً عیناً همین بسته را پس بفرستید!
Echo Reply	I ← R	بسته پس فرستاده شد! (پاسخ بسته Echo Request)
Discard Request	I → R	لطفاً این بسته را ندیده بگیرید. (حذف کنید.)
Protocol Reject	I ← R	پروتکلی را تعیین کرده‌اید که تشخیص داده نمی‌شود.

۳) استانداردهای واسط شبکه‌های محلی با کانال اشتراکی

استانداردهای انتقال اطلاعات بر روی کانال مشترک و مدیریت کانال

استانداردهای سری IEEE 802.X

۳-۱) IEEE 802.3 : استاندارد شبکه‌های محلی باس

• تعریف این استاندارد برای شبکه‌های کانال مشترک با توپولوژی باس

• مدیریت کانال به روش **CSMA/CD** : **Carrier Sense Multiple Access / Collision Detection**

روش CSMA/CD:

- گوش دادن ایستگاه متقاضی ارسال فریم به کانال
- در صورت آزاد بودن کانال آغاز ارسال فریم
- اشغال بودن کانال توسط ایستگاه دیگر ← منتظر شدن تا اتمام ارسال و در صورت آزاد شدن کانال شروع ارسال فریم ← احتمال تصادم سیگنال به دلیل منتظر بودن ایستگاههای دیگر جهت ارسال فریم
- جهت کشف سریع تصادم : گوش دادن به کانال هنگام ارسال فریم تا در صورت بروز تصادم ارسال فریم متوقف گردد
- مواجه شدن ایستگاه آغازکننده ارسال با تصادم — تولید عدد تصادفی توسط ایستگاه و توقف ارسال فریم به مدت عدد تصادفی و گوش دادن به خط
- تولید سیگنال نویز روی کانال هنگام آگاهی هر ایستگاه از تصادم جهت اطلاع ایستگاههای دیگر

راندمان کانال در استاندارد IEEE 802.3

- F : طول فریم بر حسب بیت
- B : پهنای باند کانال
- C : سرعت انتشار
- L : طول کانال
- e : عدد نپرین (2.718.....)

$$\text{راندمان کانال} = \frac{1}{\frac{1 + 2 e.B.L}{C.F}}$$

- کاهش طول فریم ← کاهش راندمان کانال
- افزایش طول کانال ← کاهش راندمان کانال
- افزایش نرخ ارسال ← کاهش راندمان کانال

مشخصات فیزیکی استاندارد IEEE 802.3

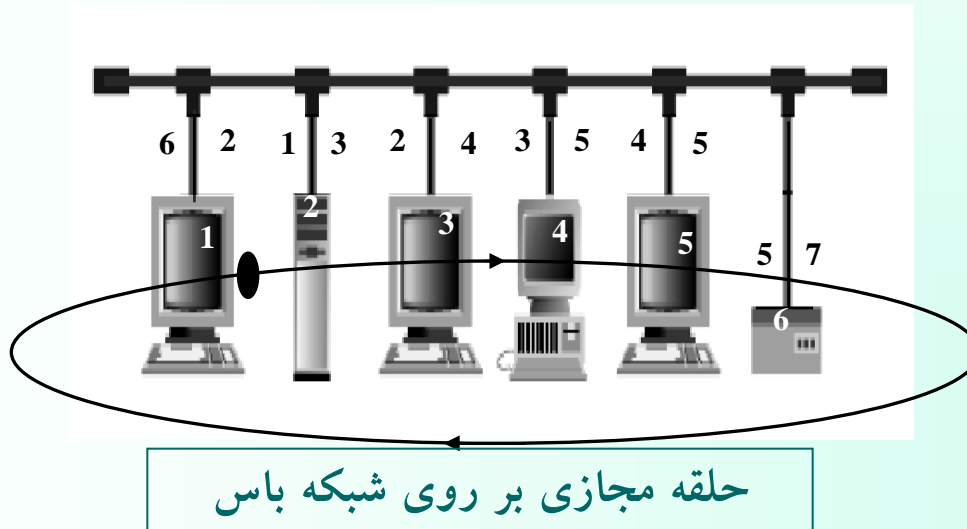
- سرعت : ۱۰ مگابیت بر ثانیه
- کدینگ : "منچستر"
- سطوح ولتاژ : 0.85 V _ و +
- کانال : کابل کوآکس ۵۰ اهم یا زوج سیم
- حداکثر طول کانال : ۵۰۰ متر با کابل کوآکس ضخیم و ۱۸۵ متر با کابل کوآکس نازک و ۱۰۰ متر با زوج سیم.

۲) IEEE 802.4 : استاندارد شبکه‌های محلی توکن باس

- هدف اصلی، پیاده‌سازی یک حلقه مجازی بر روی یک شبکه با توپولوژی باس به گونه‌ای که تصادم بر روی کانال بوجود نیاید
- استفاده همه ایستگاهها از کانال طبق یک روش سازمان‌یافته و حذف زمان تلف شده هنگام بروز تصادم
- تخمین زمان انتظار برای استفاده از کانال و ارسال فریم (اگر n ایستگاه در شبکه موجود و فعال باشد و هر ایستگاه فقط حق استفاده حداکثر T ثانیه از کانال را داشته باشد ، در بالاترین حد ترافیک ، تاخیر حداکثر $n.T$ ثانیه خواهد بود.)

روش کار:

- مطلع بودن هر ایستگاه از آدرس ایستگاه چپ و راست خود در حلقه
- ارسال یک فریم کنترلی به نام توکن به ایستگاه بعدی در حلقه بعد از اتمام ارسال فریم توسط ایستگاه
- مجوز ارسال فریم بر روی کانال در صورت داشتن فریم کنترلی توکن
- عدم بروز تصادم

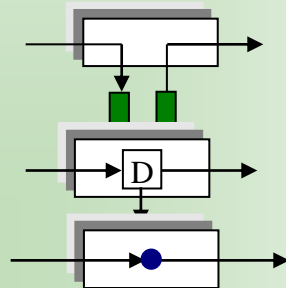


مشخصات استاندارد IEEE 802.4 :

- پیاده سازی بسیار پیچیده
- نیاز به حداقل ۱۰ زمانسنج جهت کنترل و نظارت بر استاندارد
- نوع کانال : کابل کوآکس ۷۵ اهم تلویزیون
- وجود سطوح اولویت ۰ ، ۲ ، ۴ و ۶ وبالاترین سطح اولویت ۶

۳- IEEE 802.5: استاندارد شبکه‌های محلی حلقه

- مختص توپولوژی حلقه
- دریافت فریمهای داده از ایستگاه قبلی و ارسال آنها به ایستگاه بعدی
- دریافت فریم ارسالی هر ایستگاه توسط آن ایستگاه در نهایت
- تقویت و انتقال فریم توسط ایستگاههای میانی
- ایجاد تأخیر حداقل یک بیت هنگام انتقال یک فریم توسط هر ایستگاه
- حالات ممکن هر ایستگاه:

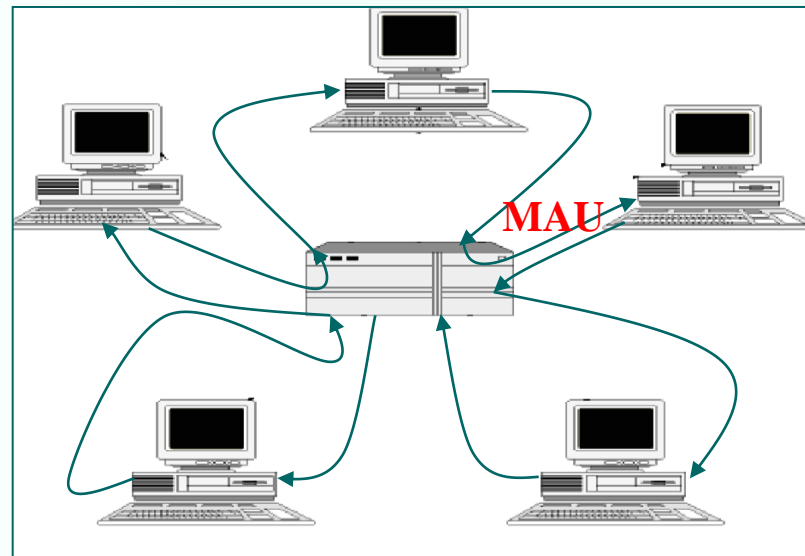


- حالت ارسال
- حالت شنود
- حالت غیرفعال

مختل شدن کل حلقه در صورت خراب شدن یکی از ایستگاهها در شبکه حلقوی

راه حل: استفاده از ابزار **MAU**

- اتصال تمام کابلهای شبکه از طریق **MAU**
- هنگام خرابی یک ایستگاه، ورودی و خروجی آن ایستگاه توسط **MAU** اتصال کوتاه می گردد.



شبکه حلقه با **MAU**: Multi Access Unit

مقایسه سه استاندارد معرفی شده برای شبکه‌های محلی

۱

IEEE 802.3 - CSMA/CD

- عدم وجود قطعیت و روال منظم در دسترسی به کانال
- وجود تأخیر بسیار کم در بار پایین و راندمان کانال مناسب
- راندمان پایین در بار بالا به دلیل افزایش تصادم
- کاهش راندمان کانال در سرعت بالا و کاهش طول فریم
- عدم وجود سطوح اولویت فریمها و ارسال صوت و تصویر در آن
- هزینه کم نصب و راه‌اندازی این نوع شبکه

IEEE 802.4 – Token Bus

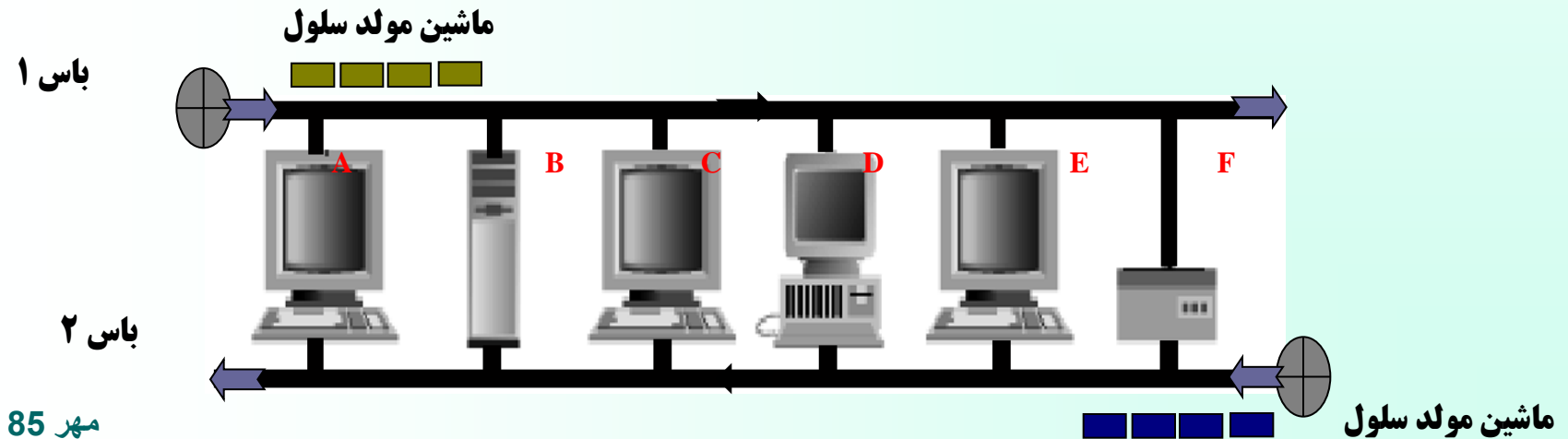
- وجود روال منظم‌تری نسبت به استاندارد **IEEE 802.3** در دسترسی به کانال.
- اولویت‌بندی فریمها و امکان ارسال همزمان و بلادرنگ صوت و تصویر در اولویت بالا
- پیچیده بودن استاندارد در اولویت بالا و آنالوگ بودن قسمتی از سخت افزار
- استفاده صحیحتر از کانال در بار بالا و با راندمان بهتر
- راندمان پائین برای فریمهای با طول کوتاه.
- قابل استفاده جهت سیستمهای بلادرنگ

IEEE 802.5 – Token Ring

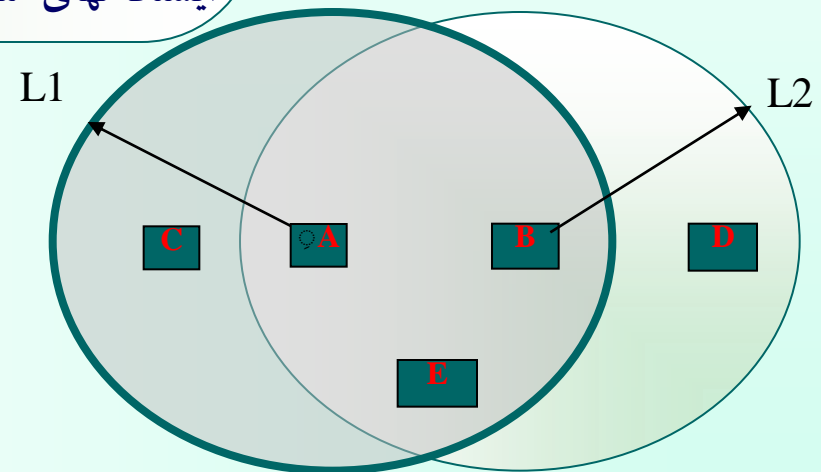
- سخت افزار کاملاً دیجیتال و عدم امکان تصادم.
- استفاده از کابل‌های زوج سیم یا فیبر نوری.
- اولویت‌بندی برای فریم‌ها و امکان ارسال همزمان و بلادرنگ صوت و تصویر با اولویت بالا
- قابلیت ارسال فریم‌های کوتاه بدون کم‌شدن راندمان کانال بصورت بحرانی
- راندمان بسیار عالی در بار بالا. (نزدیک ۱۰۰٪)
- تأثیر عملکرد بد ایستگاه ناظر بر روی کل شبکه
- وجود تأخیر ناچیز در بار پایین. (حداقل معادل زمان ۲۴ بیت)

IEEE 802.6 - DQDB : استاندارد شبکه بین شهری

- بهترین کانال انتقال برای شبکه بین شهری = فیبر نوری
- استاندارد **DQDB** مبتنی بر دو رشته فیبر نوری
- پوشش ناحیه ای به وسعت ۱۶۰ کیلومتر با نرخ ارسال **44.736Mbps** در شبکه مبتنی بر این استاندارد
- برقراری ارتباط بین ایستگاهها از طریق دو رشته فیبر نوری با طول بسیار زیاد به نام باس
- تولید سلولهای مشخص و ثابت ۵۳ بیتی به طور دائم توسط ماشینهای مولد سلول
- یکطرفه بودن مسیر و جهت ارسال اطلاعات در هر یک از باسها
- تقویت و ارسال بیتهای سلول دریافتی به قطعه بعدی توسط هر ایستگاه



- انتقال داده‌ها توسط ایستگاههای متحرک (همانند کامپیوترهای کیفی) در بُرد محدود (در حد چند ده متر) روی باند **UHF**
- وجود تعدادی ایستگاه ثابت در محدوده پیاده‌سازی چنین شبکه‌ای (ارتباط آنها نیز با ایستگاههای متحرک بی سیم است.)
- پهنای باند کانال بین یک تا دوماهگابیت بر ثانیه
- توان انتقال ثابت و محدود ایستگاههای متحرک (یعنی بُرد سیگنال تمام ایستگاهها یکسان است)
- به دلیل پراکندگی تصادفی ایستگاهها، فقط تعداد محدودی از ایستگاههای متحرک در محدوده بُرد یکدیگر هستند.



پراکندگی اتفاقی ایستگاهها در شبکه بی سیم

عملیات دست تکانی

انجام عملیات دست تکانی قبل از ارسال روی کانال توسط ایستگاهها در استاندارد **IEEE 802.11**

ارسال فریم کوتاه **RTS (Request To Send)** ۳۰ بایتی توسط ارسال کننده فریم در محدوده برد خود

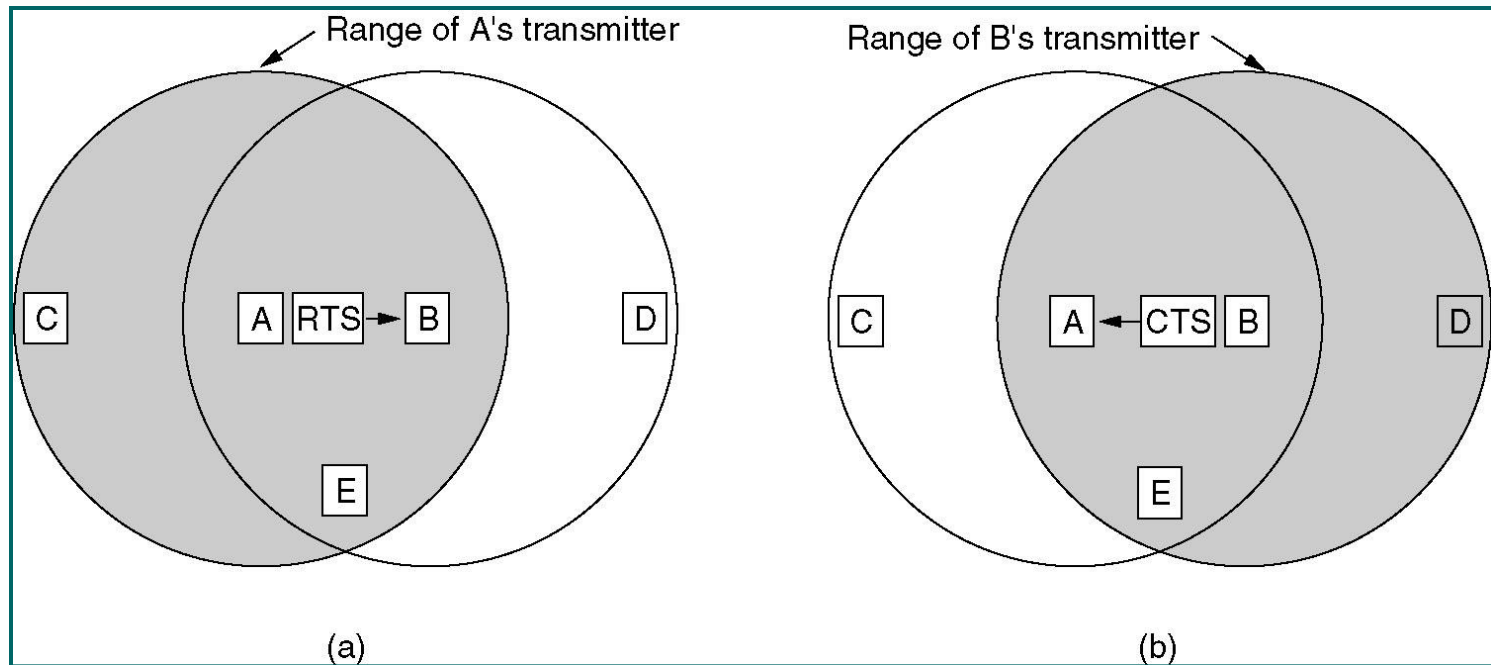
فریم **RTS** شامل: آدرس گیرنده، فرستنده و طول فریم ارسالی

ارسال فریم **CTS (Clear To Send)** در صورت آماده بودن گیرنده در پاسخ

هر ایستگاهی که سیگنال **RTS** را احساس می کند به فرستنده نزدیک است در نتیجه باید به مدت کافی صبر کند تا **CTS** بدون تصادم به فرستنده برگردد.

هر ایستگاهی که **CTS** را می شنود به گیرنده نزدیک است و باید به اندازه مدت انتقال فریم داده صبر کند تا انتقال فریم تمام شود. (طول فریم در **RTS** و **CTS** به همه ایستگاهها اعلام می شود)

ارسال فریم RTS از طرف ایستگاه A به B
برگشت فریم CTS از طرف ایستگاه B به A



استاندارد IEEE 802.11

- متغیر بودن توپولوژی شبکه
- انجام مسیریابی جهت برقراری ارتباط بین ایستگاههایی که در محدوده برد یکدیگر نیستند
- وقوع تصادم در حین ارسال فریمهای RTS و CTS

فصل سوم: لایه IP در شبکه اینترنت

اهداف آموزشی:



- مفاهیم لایه IP
- تشریح پروتکل و بسته‌های IP
- آدرس‌دهی ماشینها و کلاسهای آدرس
- الگوهای زیر شبکه
- پروتکل ICMP
- پروتکل‌های BOOTP, RARP, ARP

لایه IP

هدایت بسته‌های اطلاعاتی از شبکه‌ای به شبکه‌های دیگر

آدرسهای MAC

☺ آدرسهای قابل تعریف در لایه اول (لایه فیزیکی) جهت انتقال فریمها روی کانال

☺ وابسته به ساختار شبکه

در پروتکل CSMA/CD شبکه
MAC (Ethernet) آدرس = ۶ بایت

در پروتکل SLIP فیلد
آدرس MAC وجود ندارد

- تعریف آدرسهای جهانی و استاندارد برای تمامی ایستگاهها
- ساختار یکسان بسته قرار گرفته درون فیلد داده از فریم هر شبکه
- عدم وابستگی بسته به نوع شبکه و سخت افزار

- بی نظمی در شبکه‌های مختلف
- تنوع توپولوژی و پروتکلها
- تفاوت در روشهای آدرس دهی

بسته IP

واحد اطلاعاتی که درون فیلد داده از فریم فیزیکی قرار گرفته و با عبور از یک شبکه به شبکه دیگر تغییر نمی کند.

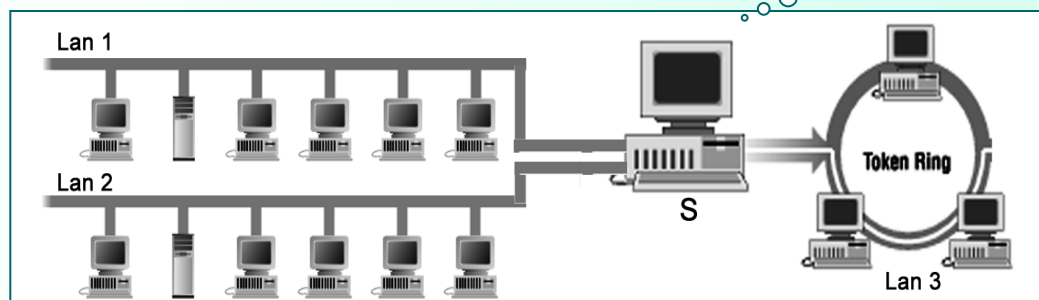
آدرس IP

آدرس جهانی و مشخص کننده ماشین به صورت یکتا و فارغ از ساختار شبکه‌ای

مسیریاب (Router)

- ماشینی با تعدادی ورودی و خروجی
- دریافت بسته‌های اطلاعاتی از ورودی و هدایت و انتخاب کانال خروجی مناسب بر اساس آدرس مقصد

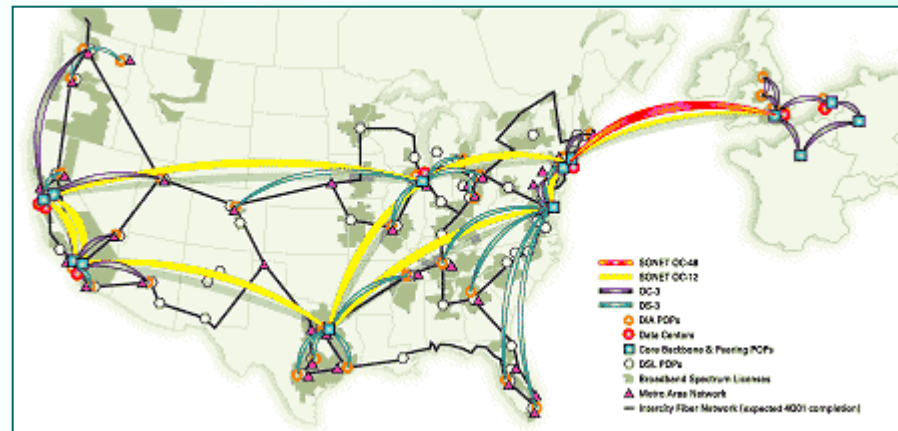
مسیریاب



لایه اینترنت (Network)

زیر شبکه (Subnet) : زیر ساخت ارتباطی شبکه ها

ستون فقرات (Backbone) : خطوط ارتباطی با پهنای باند (نرخ ارسال) بسیار بالا و مسیریابهای بسیار سریع و هوشمند در قسمت زیر شبکه



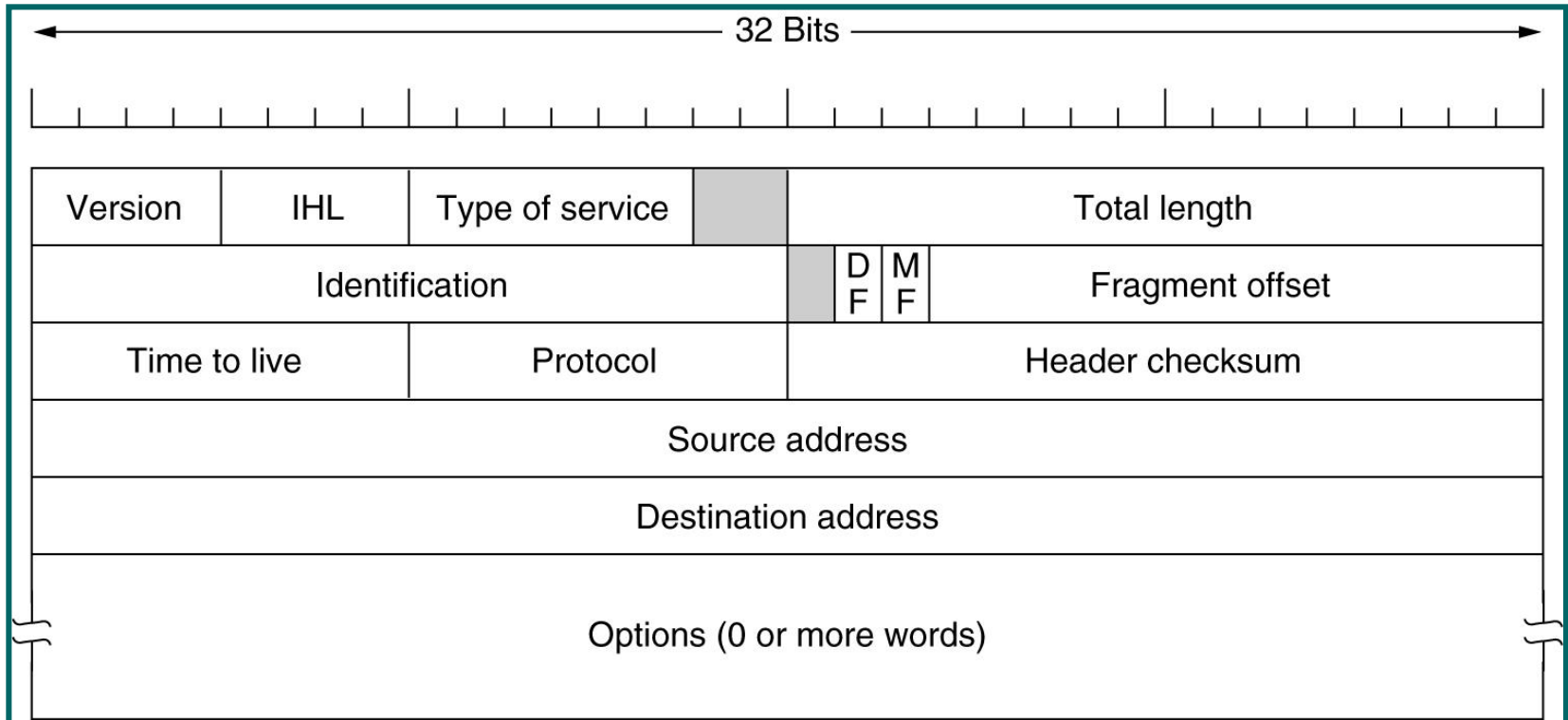
پروتکل IP:

- قرارداد حمل و تردد بسته‌های اطلاعاتی
- مدیریت و سازماندهی مسیریابی صحیح بسته‌ها از مبدأ به مقصد

دیتاگرام

واحد اطلاعات که به صورت یکجا از لایه IP به لایه انتقال تحویل داده می‌شود یا بالعکس لایه انتقال آنرا جهت ارسال روی شبکه به لایه IP تحویل داده و ممکن است شکسته شود.

قالب بسته IP



فیلد Version

- چهار بیت

- مشخص کننده نسخه پروتکل IP

نسخه شماره 4 پروتکل IP Version= 0100

نسخه شماره 6 پروتکل IP

فیلد IHL (IP Header Length)

- چهار بیتی

- مشخص کننده طول کل سرآیند بسته بر مبنای کلمات ۳۲ بیتی

- حداقل مقدار فیلد IHL عدد ۵

- فیلد ۸ بیتی
- مشخص کننده درخواست سرویس ویژه‌ای توسط ماشین میزبان از مجموعه زیرشبکه برای ارسال دیتاگرام

تعیین کننده
اولویت بسته IP

بخشهای فیلد:

P2	P1	P0	D	T	R	-	-
تقدم بسته			تاخیر	توان خروجی	قابلیت اطمینان	بلااستفاده	

قراردادن عدد ۱ توسط ماشین میزبان در این بیتها جهت انتخاب مسیر مناسب توسط مسیریابها

فیلد Total Length

- فیلد ۱۶ بیتی
- مشخص کننده طول کل بسته IP (مجموع اندازه سرآیند و ناحیه داده)
- حداکثر طول کل بسته IP ۶۵۵۳۵ بایت

فیلد Identification

- فیلد ۱۶ بیتی
- مشخص کننده شماره یک دیتاگرام واحد

فیلد Fragment Offset

الف) بیت DF (Don't Fragment):

با یک شدن این بیت در یک بسته IP هیچ مسیریابی اجازه قطعه قطعه نمودن بسته را ندارد

ب) بیت MF (More Fragment):

MF=0 : مشخص کننده آخرین قطعه IP از یک دیتاگرام
MF=1 : وجود قطعات بعدی از یک دیتاگرام

ج) Fragment offset

○ ۱۳ بیتی
○ نشان دهنده شماره ترتیب هر قطعه از یک دیتاگرام شکسته شده
○ حداکثر تعداد قطعات یک دیتاگرام ۸۱۹۲

فیلد Time To Live

- فیلد ۸ بیتی
- مشخص کننده طول عمر بسته IP
- حداکثر طول عمر بسته IP = ۲۵۵

فیلد پروتکل

- نشان دهنده شماره پروتکل لایه بالاتر متقاضی ارسال دیتاگرام
- فیلد ۸ بیتی

فیلد Header Ckecksum

- فیلد ۱۶ بیتی
- کشف خطاهای احتمالی در سرآیند هر بسته IP

روش محاسبه کد کشف خطا:

جمع کل سرآیند به صورت دو بایت دو بایت
حاصل جمع به روش مکمل یک منفی می گردد
قرارگرفتن عدد منفی حاصله در فیلد **Header Ckecksum**

فیلد Source Address

- فیلد ۳۲ بیتی
- مشخص کننده آدرس ماشین مبدأ

فیلد Destination Address

- فیلد ۳۲ بیتی
- مشخص کننده آدرس IP ماشین مقصد

فیلد Payload

قرارگرفتن داده های دریافتی از لایه بالاتر در این فیلد

فیلد اختیاری Option

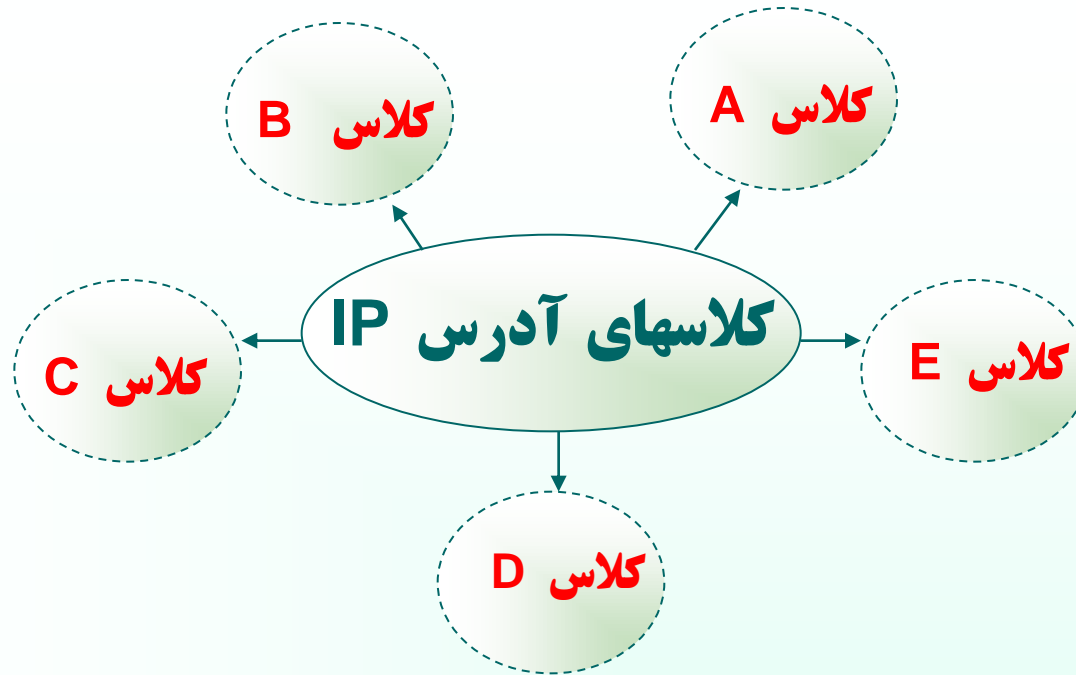
- حداکثر ۴۰ بایت
- محتوی اطلاعات جهت یافتن مسیر مناسب توسط مسیریابها

آدرسها در اینترنت و اینترانت

شناسایی تمام ابزار شبکه (ماشینهای میزبان, مسیریابها, چاپگرهای شبکه) در اینترنت با یک آدرس IP

آدرس IP

- ۳۲ بیتی
- پرارزشتین بایت آدرس IP مشخص کننده کلاس آدرس
- نوشتن آدرسهای IP به صورت چهار عدد دهمی که با نقطه از هم جدا شده اند جهت سادگی نمایش



تقسیم ۳۲ بیت آدرس IP به قسمتهای :
آدرس ماشین / آدرس زیر شبکه / آدرس شبکه

آدرسهای کلاس A

- مقدر را پر ارزشترین بیت = ۰
- ۷ بیت از یک بایت اول = مشخصه آدرس IP
- ۳ بایت باقیمانده مشخص کننده آدرس ماشین میزبان
- بایت پر ارزش در محدوده صفر تا ۱۲۷

Network ID = ۷ Bit



1.0.0.0 to
127.255.255.255

کلاس B

- مقدار دو بیت پر ارزش = ۱۰
- ۱۴ بیت از دو بایت سمت چپ = آدرس شبکه
- دو بایت اول از سمت راست = آدرس ماشین میزبان

Network ID = 14 Bit



192.0.0.0 to
239.255.255.255

کلاس C

- مناسب‌ترین و پرکاربردترین کلاس از آدرسهای IP
- مقدار سه بیت پرارزش = ۱۱۰
- ۲۱ بیت از سه بایت سمت چپ = مشخص‌کننده آدرس شبکه
- ۸ بیت سمت چپ = آدرس ماشین میزبان



کلاس D

- مقدار چهار بیت پرارزش = ۱۱۱۰
- ۲۸ بیت = تعیین آدرسهای چند مقصده (آدرسهای گروهی)
- کاربرد = عملیات رسانه‌ای و چند بخشی



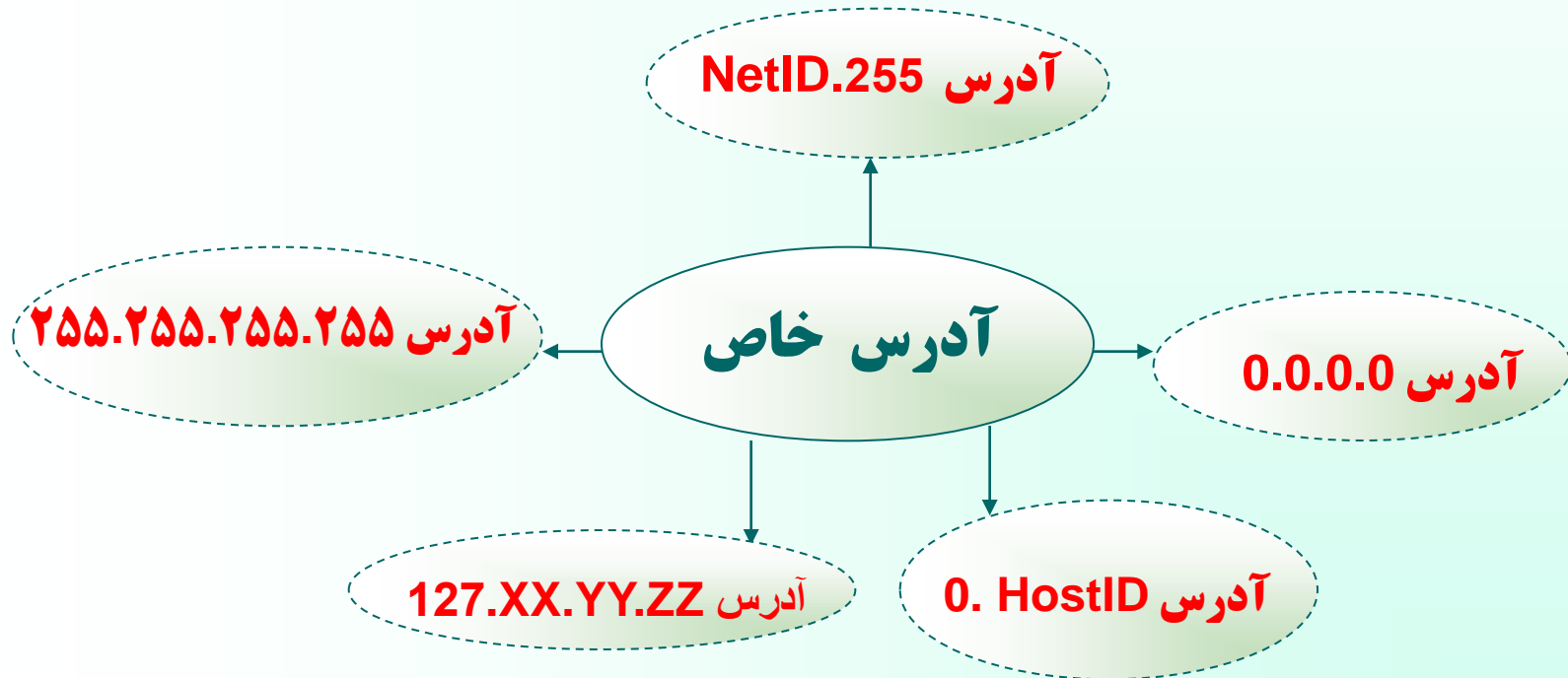
کلاس E

• مقدار پنج بیت پر ارزش = ۱۱۱۱۰



آدرسهای خاص

در بین تمام کلاسهای آدرس IP با پنج گروه از آدرسها نمی توان یک شبکه خاص را تعریف و آدرس دهی نمود.



آدرس 0.0.0.0:

هر ماشین میزبان که از آدرس IP خودش مطلع نیست این آدرس را بعنوان آدرس خودش فرض می کند.

آدرس HostID 0.0.0.0 :

این آدرس زمانی به کار می رود که ماشین میزبان ، آدرس مشخصه شبکه ای که بدان متعلق است را نداند. در این حالت در قسمت NetID مقدار صفر و در قسمت HostID شماره مشخصه ماشین خود را قرار می دهد.

0 0		This host		
0 0	...	0 0	Host	A host on this network
1 1				Broadcast on the local network
Network	1 1 1 1	...	1 1 1 1	Broadcast on a distant network
127	(Anything)			Loopback

آدرس 255.255.255.255:

جهت ارسال پیامهای فراگیر برای تمامی ماشینهای میزبان بر روی شبکه محلی که ماشین ارسال کننده به آن متعلق است .

آدرس NetID.255 :

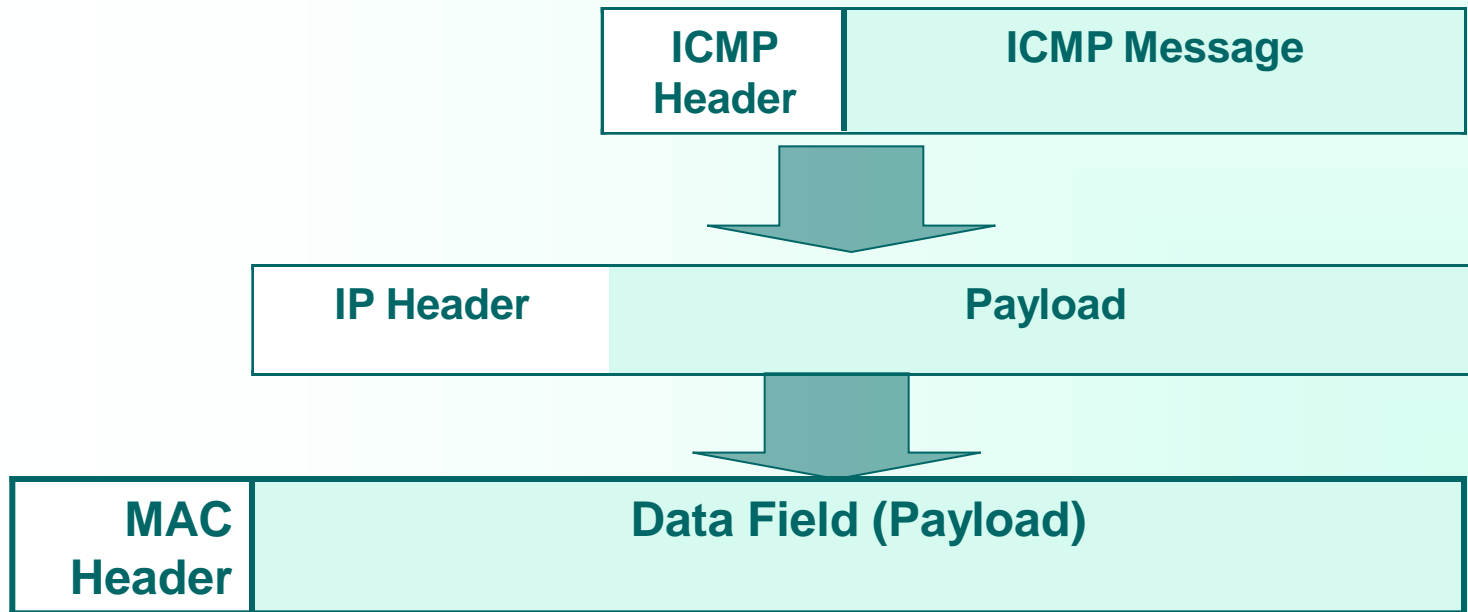
جهت ارسال پیامهای فراگیر برای تمامی ماشینهای یک شبکه راه دور که ماشین میزبان فعلی متعلق به آن نیست .

آدرس 127.xx.yy.zz:

این آدرس بعنوان "آدرس بازگشت" شناخته می شود و آدرس بسیار مفیدی برای اشکالزدایی از نرم افزار می باشد .

پروتکل ICMP: Internet Control Message Protocol

- بررسی انواع خطا و ارسال پیام برای مبدأ بسته در صورت بروز خطا و اعلام نوع خطا
- یک سیستم گزارش خطا
- قرارگرفتن پیام ICMP درون بسته IP



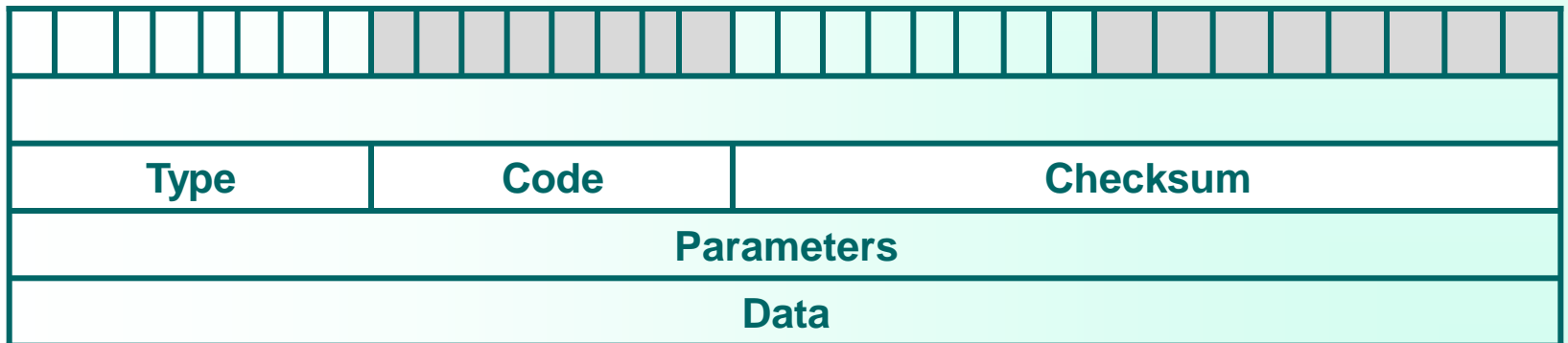
قالب پیام ICMP

فیلد **Type**: مشخص کننده نوع پیام

فیلد **Code**: مشخص کننده کد زیرنوع

فیلد **Checksum**: جهت سنجش اعتبار و درستی بسته ICMP

۳۲ بیت



انواع پیامهای ICMP

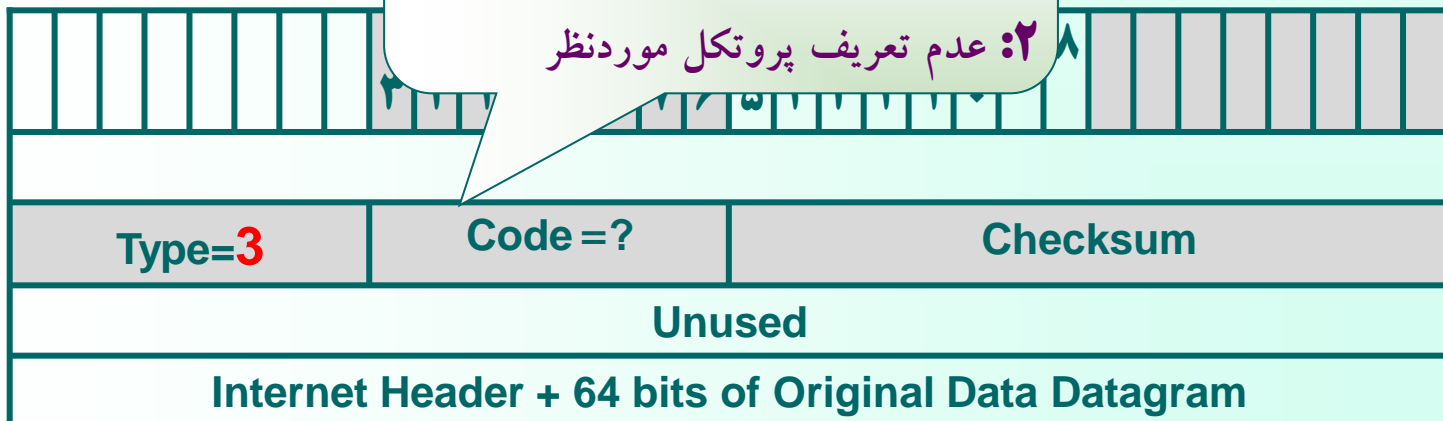
1) پیام Destination Unreachable

- عدم تشخیص آدرس توسط مسیریاب و یا زیر شبکه
- نرسیدن بسته به مقصد به هر علت

♦ در دسترس نبودن شبکه مورد نظر

۱: در دسترس نبودن ماشین میزبان

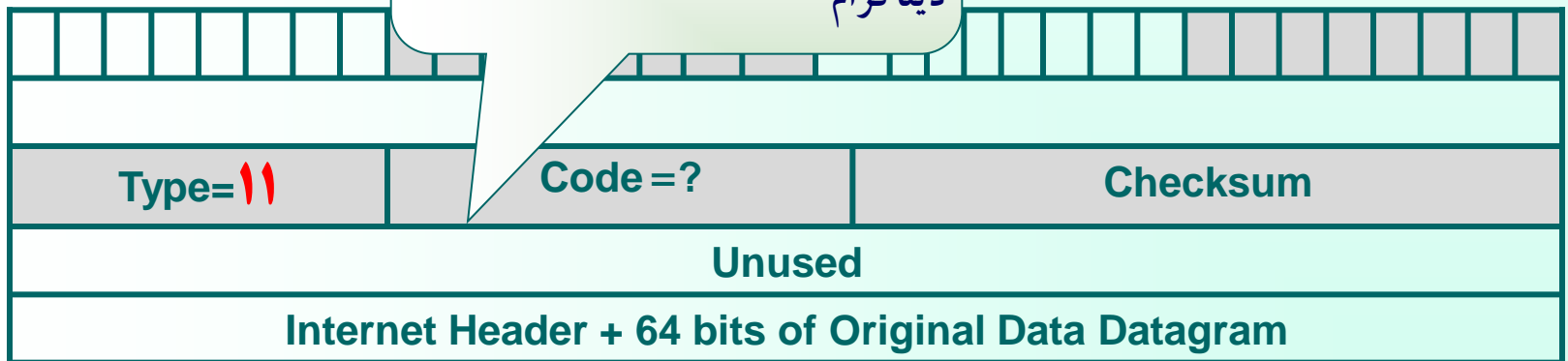
۲: عدم تعریف پروتکل موردنظر



٢) پیام Time Exceeded

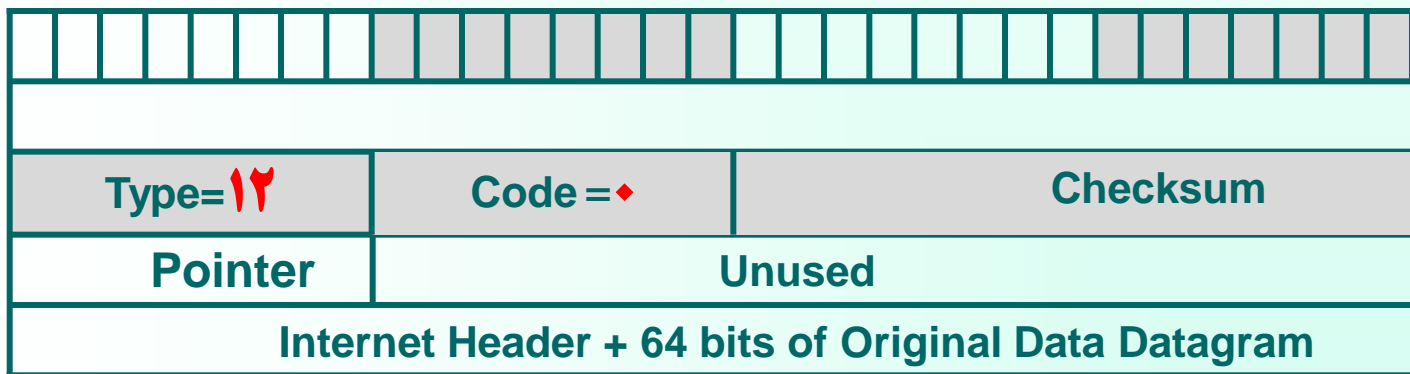
ارسال پیام به فرستنده بسته جهت آگاهی از اتمام طول عمر بسته و حذف آن توسط مسیریاب

• = اتمام زمان حیات بسته
۱ = اتمام زمان بازسازی قطعات یک دیتاگرام



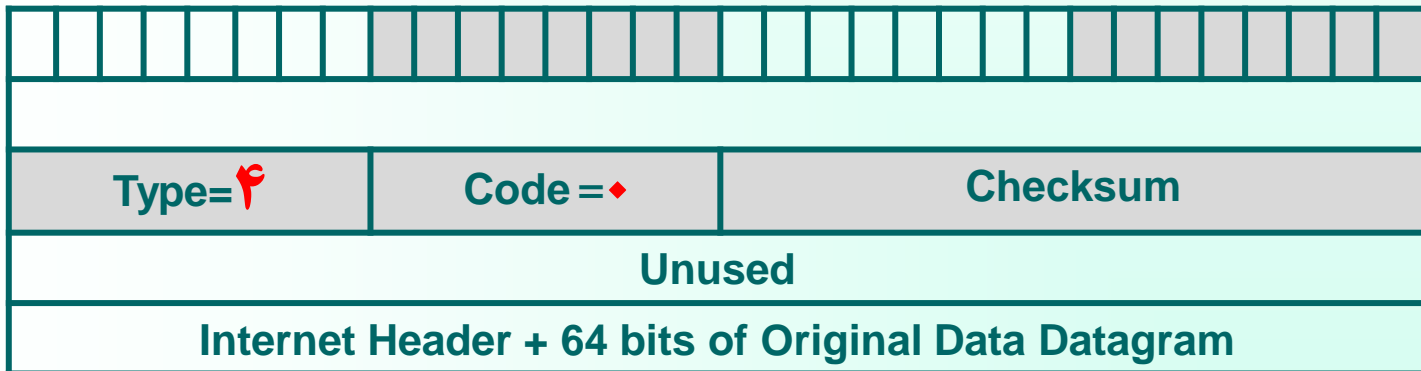
۳) پیام Parameter Problem

نشان‌دهنده وجود مقدار نامعتبر در یکی از فیلدهای سرآیند بسته
IP



Source Quench پیام (۴)

تقاضای کاهش نرخ تولید و ارسال بسته‌های IP از ماشین میزبان



۵) پیام Redirect

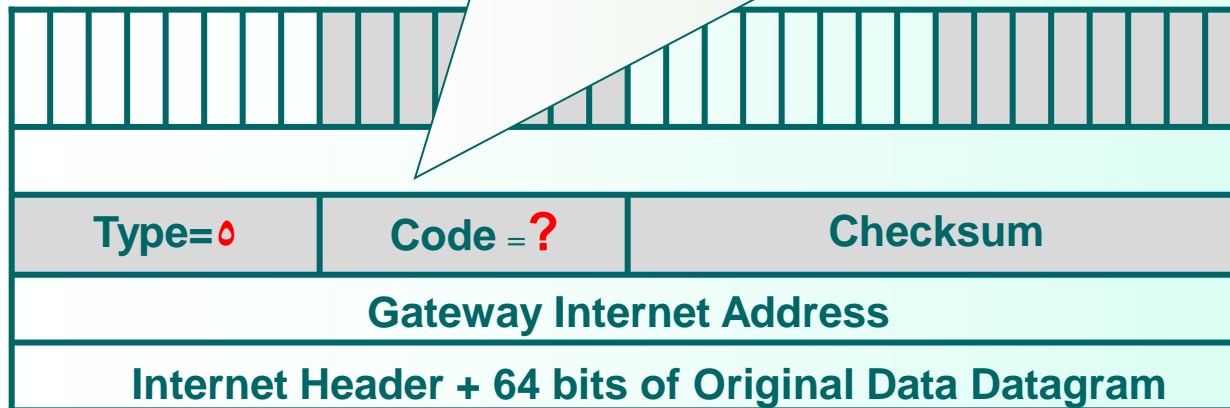
وجود اشکال در مسیریابی

۰ = تغییر مسیر به شبکه‌ای که آدرس آن مشخص شده است.

۱ = تغییر مسیر به ماشینی که آدرس آن مشخص شده است.

۲ = تغییر مسیر به شبکه‌ای که آدرس آن مشخص شده است جهت تأمین سرویس ویژه
Type of service درخواستی مشخص شده در فیلد

۳ = تغییر مسیر به ماشینی که آدرس آن مشخص شده است جهت تأمین سرویس ویژه
Type of service درخواستی مشخص شده در فیلد

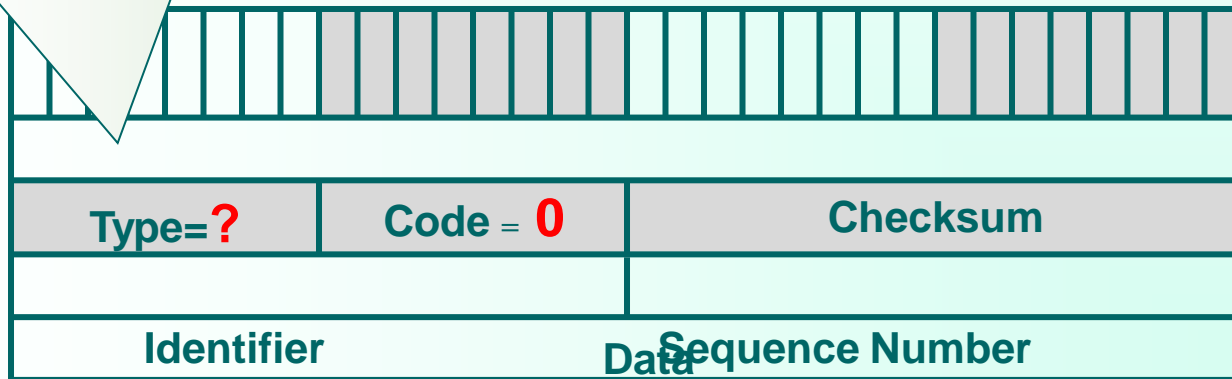


۶ پیامهای Echo Request , Echo Reply

پیام **Echo Request** : موجود و قابل دسترس بودن یک ماشین خاص در شبکه توسط مسیریاب

پیام **Echo Reply** : پاسخ مقصد مبنی بر دریافت پیام **Echo Request**

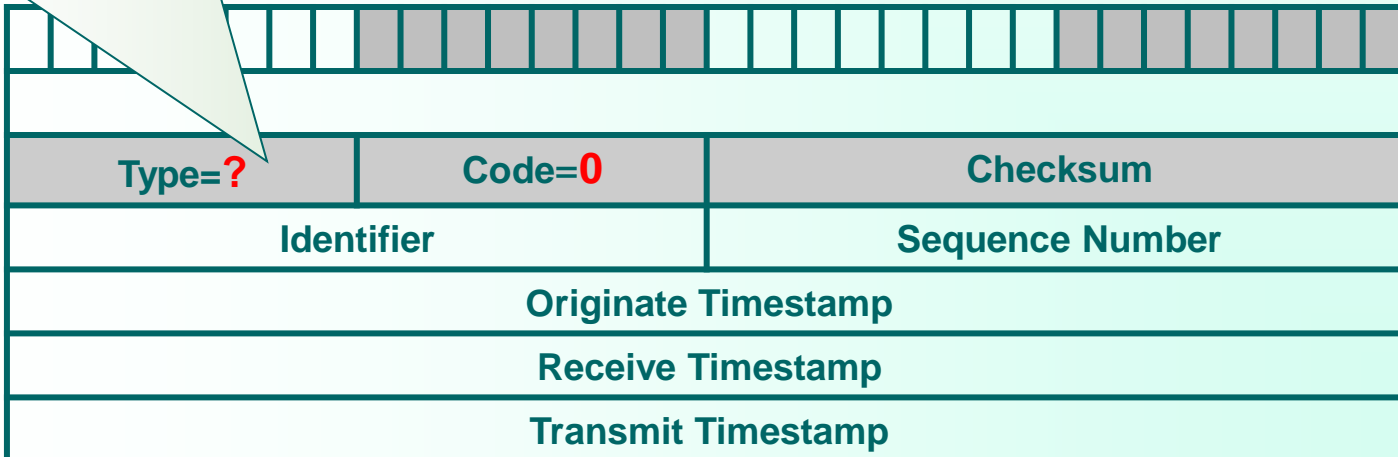
8 : برای مشخص کردن پیام **Echo Request**
0 : برای مشخص کردن پیام **Echo Reply**



پیامهای Timestamp Request و Timestamp Reply (۷)

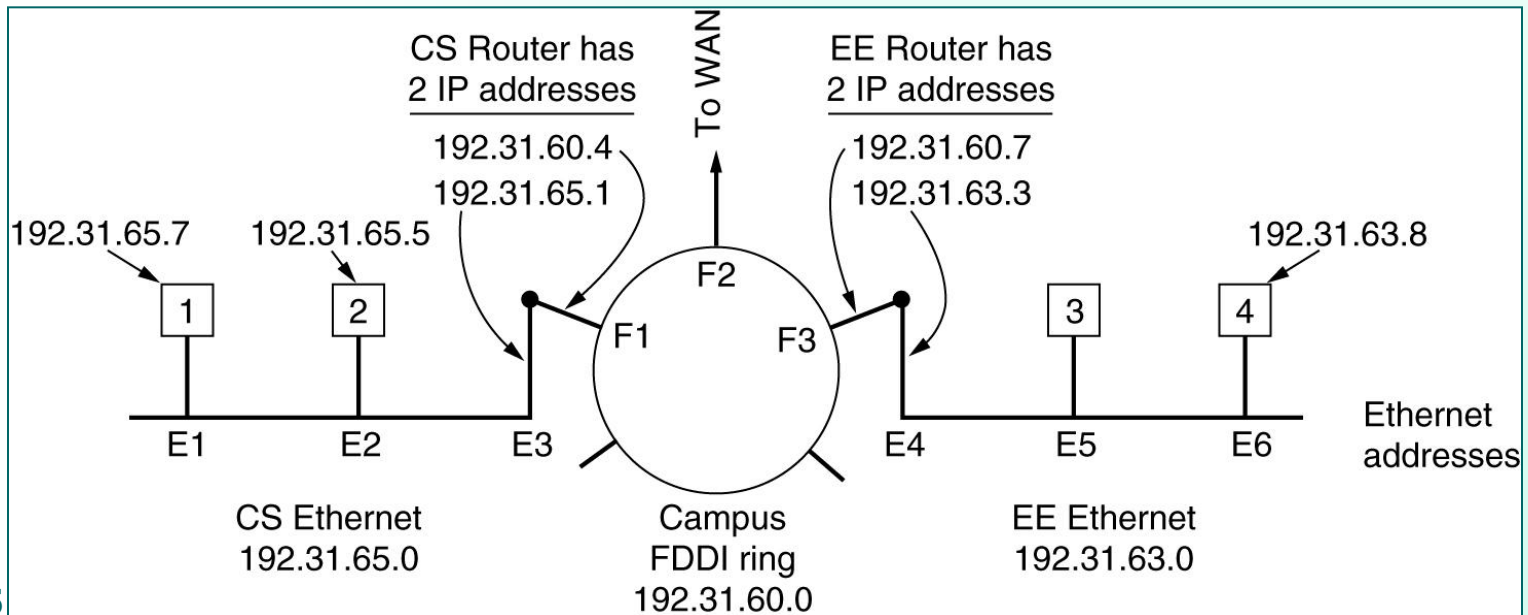
دریافت کننده پیام Timestamp Request زمان دریافت و زمان ارسال بسته را نیز مشخص می کند.

13 : برای مشخص کردن پیام Timestamp Request
14 : برای مشخص کردن پیام Timestamp Reply

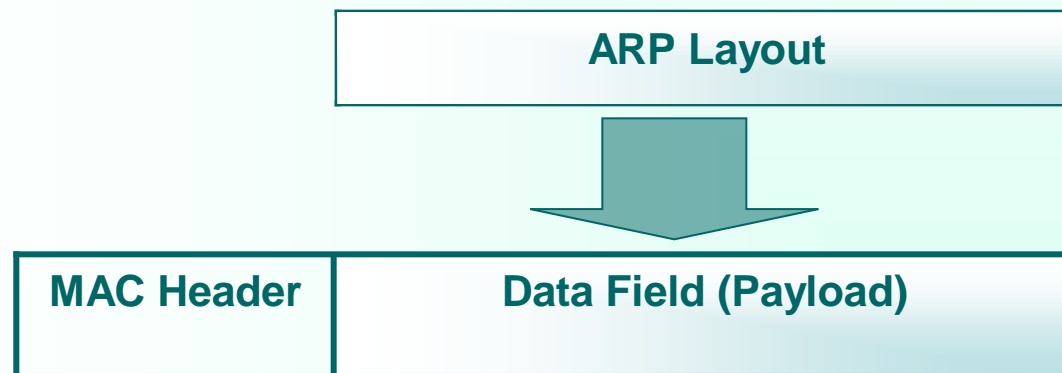


پروتکل ARP : Address Resolution Protocol

- بی‌معنابودن آدرسهای IP روی کانال انتقال
 - دانستن آدرس IP ماشین مقصد و نیاز به داشتن آدرس فیزیکی آن جهت ارسال بسته
 - وظیفه پروتکل ARP:
 - ارسال بسته فراگیر روی کل شبکه محلی که در آن آدرس IP ماشین مورد نظر قرار دارد. پاسخ ماشین با آدرس IP موجود در بسته ارسالی و ارسال آدرس فیزیکی خود برای ارسال‌کننده بسته
- ARP**



برخلاف پروتکل ICMP که روی پروتکل IP قرار می‌گیرد، پروتکل ARP مستقیماً بر روی پروتکل لایه فیزیکی عمل می‌کند؛ یعنی یک بسته ARP ساخته شده و درون فیلد داده از فریم لایه فیزیکی قرار گرفته و روی کانال ارسال می‌شود.



چگونگی قرار گرفتن یک پیام ARP درون فریم لایه فیزیکی

ساختار پیامهای ARP

Hardware Type	
Protocol Type	
Hardware Address Length	Protocol Address Length
Operation Code	
Source Hardware Address	
Source IP Address	
Destination Hardware Address	
Destination IP Address	

پروتکل RARP : Reverse Address Resolution Protocol

- ایستگاه آدرس فیزیکی مورد نظرش را می‌داند ولیکن آدرس IP آن را نمی‌داند
- ارسال یک بسته فراگیر روی خط
- تمامی ایستگاههایی که از پروتکل RARP حمایت می‌کنند و بسته‌های مربوطه را تشخیص می‌دهند، در صورتی که آدرس فیزیکی خودشان را درون بسته ببینند در پاسخ به آن، آدرس IP خود را در قالب یک بسته RARP Reply برمی‌گردانند.

توجه: بسته‌های RARP, ARP از نوع فراگیر محلی Local Broadcast هستند و بالطبع توسط مسیریابها منتقل نمی‌شوند و فقط در محدوده شبکه محلی عمل می‌کنند.

پروتکل *BootP*

- گاهی نیاز است که یک آدرس **IP** روی چند شبکه محلی جستجو شود که در این حالت **RARP** جوابگو نیست .
- داشتن آدرس فیزیکی ماشین مورد نظر و نیاز به پیدا کردن آدرس **IP** آن در شبکه‌های محلی دیگر
- استفاده از بسته‌های **UDP** در این پروتکل

فصل چهارم : مسیریابی در شبکه اینترنت

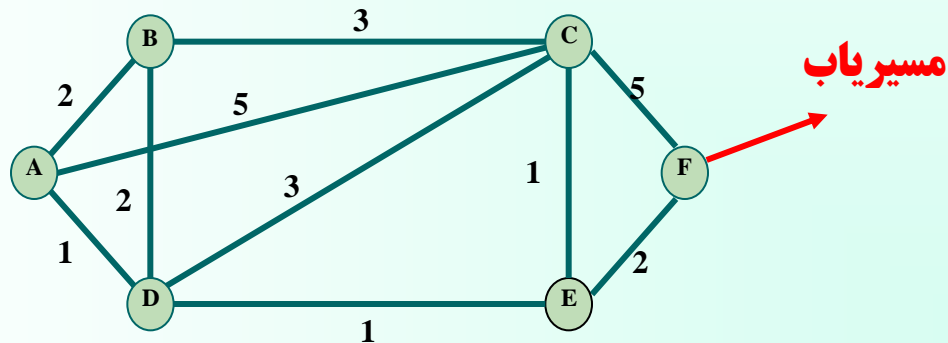
اهداف آموزشی :



- مفاهیم اولیه مسیریابی
- الگوریتم‌های مسیریابی LS
- الگوریتم‌های مسیریابی بردار فاصله - DV -
- مسیریابی سلسله مراتبی
- پروتکل RIP
- پروتکل OSPF
- پروتکل BGP

(۱) مفاهیم اولیه مسیریابی

مسیریاب: ابزاری است برای برقراری ارتباط دو یا چند شبکه
زیرساخت ارتباطی: مجموعه مسیریابها و کانالهای فیزیکی ما بین آنها
الگوریتم‌های مسیریابی: روشهایی برای پیدا کردن مسیری بهینه میان
دو مسیریاب به گونه‌ای که هزینه کل مسیر به حداقل برسد.



زیرساخت ارتباطی یک شبکه فرضی

برخی اصطلاحات کلیدی در مسیریابی

آدرسهای MAC:

- آدرسهای لایه فیزیکی جهت انتقال فریمها بر روی کانال
- اندازه آدرس وابسته به پروتکل و توپولوژی شبکه
- تغییر آدرسهای **MAC** بسته‌های اطلاعاتی هنگام عبور از مسیریابهای موجود در مسیر

آدرسهای IP:

- آدرسهای جهانی و منحصر به فرد
- مشخص کننده یک ماشین فارغ از نوع سخت افزار و نرم افزار آن
- ثابت بودن آدرسهای **IP** بسته های اطلاعاتی هنگام عبور از مسیریابهای موجود در مسیر

بسته IP:

- واحد اطلاعاتی با اندازه محدود

توپولوژی شبکه:

- مجموعه مسیریابها و کانالهای فیزیکی ما بین آنها در زیرساخت ارتباطی یک شبکه
- متغیر با زمان

ترافیک شبکه:

- تعداد متوسط بسته‌های اطلاعاتی ارسالی و یا دریافتی روی یک کانال در واحد زمان
- متغیر با زمان

گام یا Hop:

- عبور بسته از یک مسیریاب = گام
- تعداد مسیریابهای موجود در مسیر یک بسته = تعداد گام = Hop Count

ازدحام یا Congestion:

بیشتر بودن تعداد متوسط بسته‌های ورودی به یک مسیریاب از تعداد متوسط بسته‌های خروجی

بن بست Deadlock:

پایان طول عمر بسته‌ها

۱-۱) روشهای هدایت بسته‌های اطلاعاتی در شبکه‌های کامپیوتری

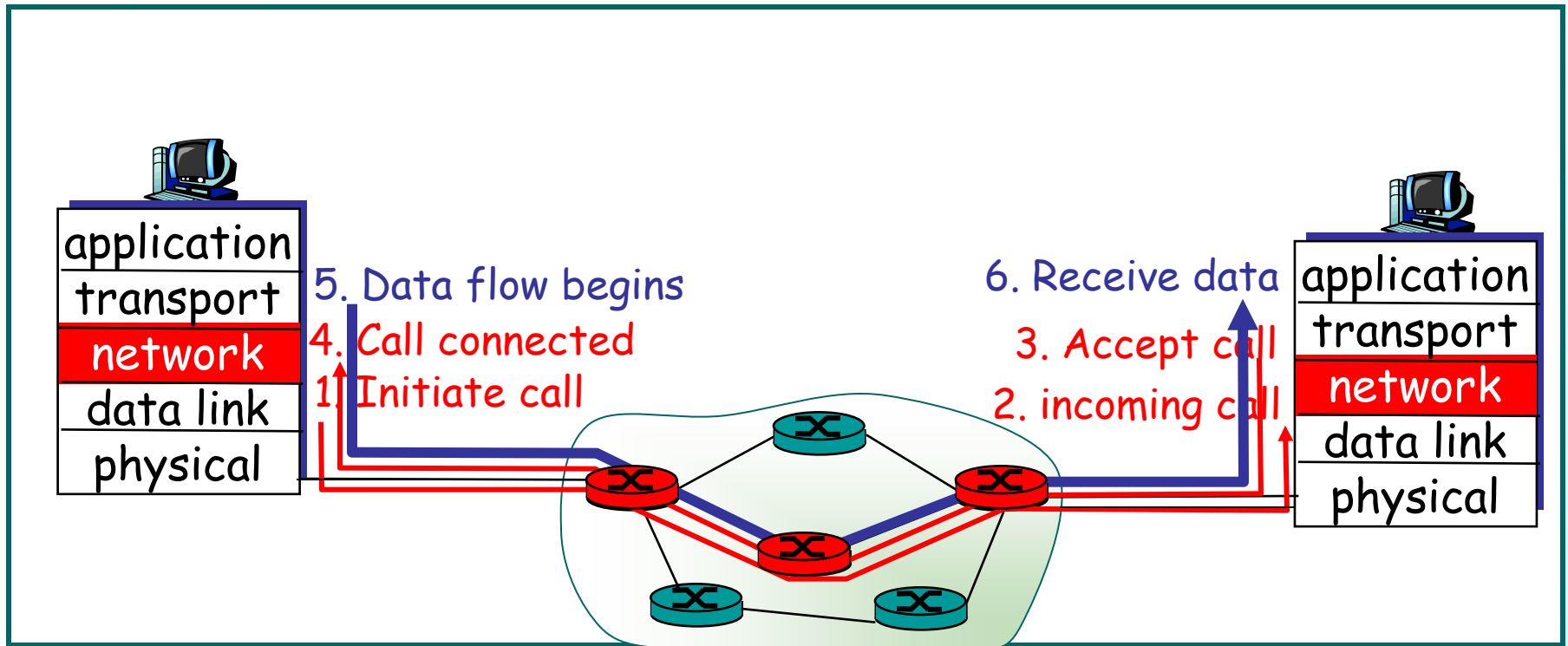
الف) روش مدار مجازی (VC) Virtual Circuit

ب) روش دیتاگرام Datagram

خصوصیات روش VC

- ارسال بسته‌های اطلاعاتی بدون نیاز به اطلاع از آدرسهای IP مبدأ و مقصد و فقط داشتن شماره VC جهت ارسال بسته
- عدم اجرای الگوریتم مسیریابی جهت هدایت بسته‌های اطلاعاتی از مبدأ به مقصد
- دریافت بسته به ترتیب ارسال شده در مقصد
- عدم احتمال گم شدن بسته‌ها در عمل مسیریابی در شبکه

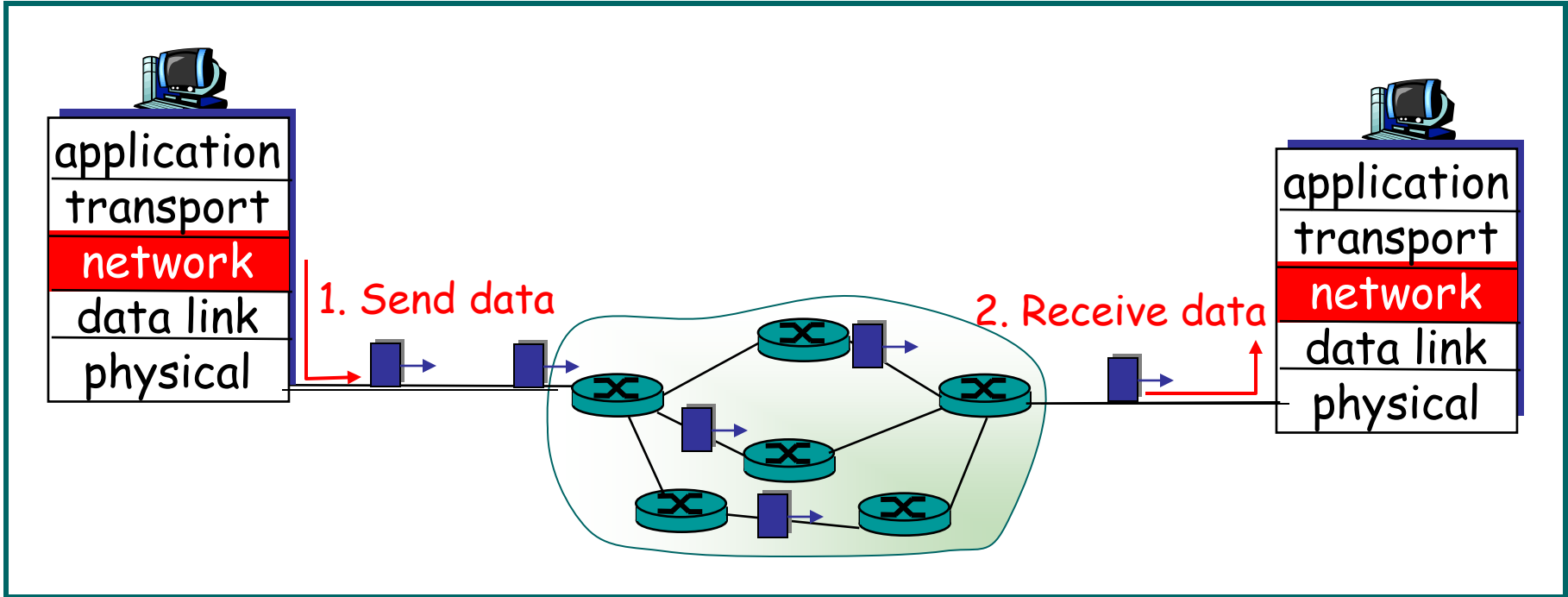
روش VC



خصوصیات روش دیتاگرام

- ارسال بسته‌های اطلاعاتی با استفاده از آدرس‌های IP مبدأ و مقصد در شبکه
- انجام مسیریابی جداگانه برای هر بسته
- توزیع و هدایت بسته‌ها روی مسیرهای متفاوت بر اساس شرایط توپولوژیکی و ترافیکی لحظه‌ای شبکه
- امکان دریافت بسته بدون ترتیب ارسال شده در مقصد
- لزوم نظارت‌های ویژه بر گم شدن و یا تکراری بودن بسته در لایه‌های بالاتر

روش Datagram



انواع الگوریتمهای مسیریابی

ب) از دیدگاه چگونگی جمع‌آوری و پردازش اطلاعات
زیرساخت ارتباطی شبکه

غیرمتمرکز

سراسری / متمرکز

الف) از دیدگاه روش تصمیم‌گیری و میزان هوشمندی
الگوریتم

پویا

ایستا

الگوریتم ایستا

- عدم توجه به شرایط توپولوژیکی و ترافیک لحظه‌ای شبکه
- جداول ثابت مسیریابی هر مسیریاب در طول زمان
- الگوریتم‌های سریع
- تنظیم جداول مسیریابی به طور دستی در صورت تغییر توپولوژی زیرساخت شبکه
- تغییر مسیرها به کندی در اثنای زمان

الگوریتم پویا

- به هنگام سازی جداول مسیریابی به صورت دوره‌ای بر اساس آخرین وضعیت توپولوژیکی و ترافیک شبکه
- تغییر سریع مسیرها
- تصمیم‌گیری بر اساس وضعیت فعلی شبکه جهت انتخاب بهترین مسیر
- ایجاد تأخیرهای بحرانی هنگام تصمیم‌گیری بهترین مسیر به جهت پیچیدگی الگوریتم

الگوریتم سراسری

- اطلاع کامل تمام مسیریابها از همبندی شبکه و هزینه هر خط
- الگوریتم‌های **Link State (LS)**

الگوریتم غیر متمرکز

- محاسبه و ارزیابی هزینه ارتباط با مسیریابهای همسایه (مسیریابهایی که به صورت مستقیم و فیزیکی با آن در ارتباط هستند)
- ارسال جداول مسیریابی توسط هر مسیریاب در فواصل زمانی منظم برای مسیریابهای مجاور
- پیچیدگی زمانی کم
- الگوریتم‌های **Distance Vector**

۳-۱) روش ارسال سیل آسا (Flooding Algorithm)

- سریعترین الگوریتم برای ارسال اطلاعات به مقصد در شبکه
- جهت ارسال بسته‌های فراگیر و کنترل‌ی مانند اعلام جداول مسیریابی

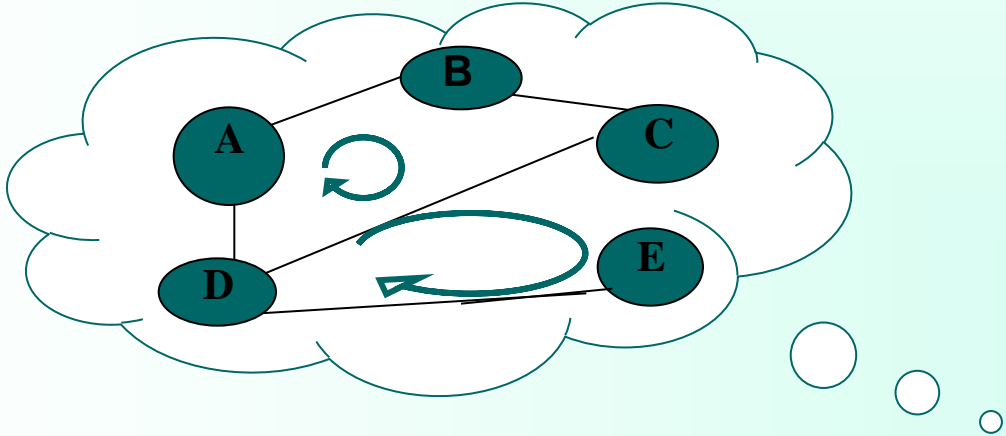
مشکل روش سیل آسا

- ایجاد حلقه بینهایت و از کارافتادن شبکه

راه حل رفع مشکل حلقه بینهایت

۱) قراردادن شماره شناسایی برای هر بسته Selective Flooding

۲) قراردادن طول عمر برای بسته‌ها



حلقه‌های بینهایت در روش سیل آسا

الگوریتم های LS

- ۱- شناسایی مسیرهای مجاور
- ۲- اندازه گیری هزینه
- ۳- تشکیل بسته های LS
- ۴- توزیع بسته های LS روی شبکه
- ۵- محاسبه مسیرهای جدید

۱- شناسایی مسیرهای مجاور

- ارسال بسته خاصی به نام بسته سلام **Hello Packet** توسط مسیریاب به تمام خروجی ها
- پاسخگویی مسیرهای متصل از طریق کانال فیزیکی مستقیم به بسته ارسالی و اعلام آدرس **IP** خود به مسیریاب
- درج اطلاعات بسته های پاسخ در جدول مسیریاب

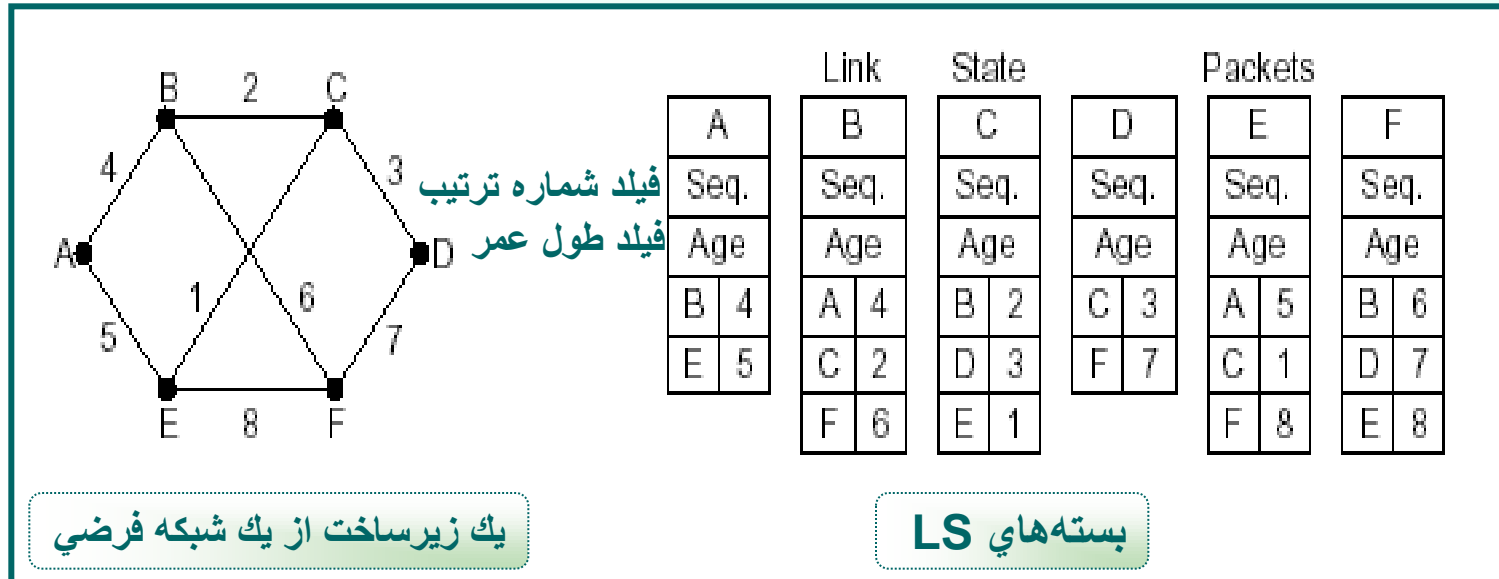
۲- اندازه‌گیری هزینه

- اندازه‌گیری تأخیر هر یک از خطوط خروجی مسیریاب توسط خود مسیریاب
- ارسال بسته خاص به نام **Echo Packet** روی تمام خطوط خروجی خود
- پاسخ تمام مسیریابهای گیرنده بسته با ارسال بسته **Echo Reply**
- اگر مسیریاب موظف باشد که با دریافت بسته **Echo** خارج از نوبت و به سرعت به آن پاسخ بدهد، "زمان رفت و برگشت" این بسته فقط تاخیر فیزیکی بین دو مسیریاب را به عنوان معیار هزینه مشخص می‌کند.
- اندازه‌گیری این زمان با استفاده از زمان سنج و تقسیم آن مقدار بر عدد ۲ و درج در جدول توسط مسیریاب

۳- تشکیل بسته‌های LS

تشکیل بسته LS پس از جمع آوری اطلاعات لازم از مسیریابهای مجاور شامل:

- الف) آدرس جهانی مسیریاب تولیدکننده بسته
 ب) یک شماره ترتیب (تا بسته‌های تکراری از بسته‌های جدید تشخیص داده شوند).
 ج) طول عمر بسته (تا اطلاعات بسته، زمان انقضای اعتبار داشته باشد).
 د) آدرس جهانی مسیریابهای مجاور و هزینه تخمینی



۴- توزیع بسته‌های LS روی شبکه

- ارسال بسته‌های LS به روش سیل آسا
- وجود شماره ترتیب برای هر بسته جهت جلوگیری از بروز حلقه تکرار
- در نظرگرفتن طول عمر برای هر بسته جهت رفع مشکل دریافت بسته‌های تکراری
- احراز هویت ارسال‌کننده بسته LS در مسیریابها جهت جلوگیری از بسته‌های LS آلوده

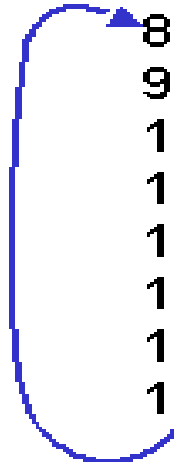
- تشکیل ساختمان داده گراف زیر شبکه جهت انتخاب بهترین مسیر بین دو گره هنگام دریافت بسته‌های **LS** از تمام مسیریابهای شبکه
- استفاده از الگوریتم دایجکسترا جهت یافتن بهترین مسیر بین دو گره

(Dijkstra Shortest Path Algorithm)

- * $C(i, j)$ بیانگر هزینه خط میان گره i تا j است.
هرگاه همسایگانی در مجاورت گره وجود نداشته باشند
- * $C(i, j)$ بینهایت تلقی می شود.
- * $D(v)$ هزینه فعلی مسیر میان مبدا تا گره v .
- * $P(v)$ گره‌ای که در طول مسیر از مبدا تا v درست قبل از v واقع شده.
- * N مجموعه گره‌هایی که عبور از آنها کم هزینه برآورد گشته است.

Dijkstra's Algorithm

```
1 Initialization:  
2   N = {A}  
3   for all nodes v  
4     if v adjacent to A  
5       then D(v) = c(A,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
```



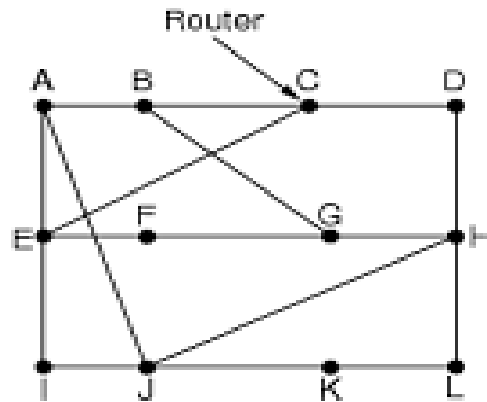
الگوریتمهای DV یا بردار فاصله

- یکی از روشهای پویا در مسیریابی
- مورد استفاده در شبکه ARPA
- استفاده در مسیریابهای کوچک
- نامهای متفاوت روش DV
- پروتکل RIP
- الگوریتم مسیریابی Bellman - Ford
- الگوریتم مسیریابی Ford – Fulkerson
- الگوریتم Distance Vector Routing

اصول کار روش DV

- محاسبه خطوطی را که به صورت فیزیکی با مسیریابهای دیگر دارد و درج در جدول مسیریابی
- بینهایت در نظر گرفتن هزینه خطوطی که مسیریاب با آنها در ارتباط مستقیم نیست
- ارسال ستون هزینه از جدول مسیریابی برای مسیریابهای مجاور در بازه‌های زمانی مشخص، توسط هر مسیریاب (“یعنی فقط برای مسیریابهایی که با آن در ارتباط است نه تمام مسیریابها”). دریافت اطلاعات جدید از مسیریابهای مجاور در در فواصل **T** ثانیه‌ای
- به هنگام نمودن جدول مسیریابی پس از دریافت جداول مسیریابی از مسیریابهای مجاور، طبق یک الگوریتم بسیار ساده

الگوریتمهای DV یا بردار فاصله



(a)

زیرساخت ارتباطی یک شبکه فرضی
با دوازده مسیر یاب

To	A	I	H	K	New estimated delay from J	
					↓ 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

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

Vectors received from J's four neighbors

New routing table for J

(b)

جدول مسیریابی مربوط به مسیر یاب J

مشکل عمده پروتکل‌های DV

عدم همگرایی سریع جداول مسیریابی هنگام خرابی یک مسیریاب یا یک کانال ارتباطی = مشکل شمارش تا بینهایت

راه حل:

وقتی یک مسیریاب می‌خواهد اطلاعاتی را به همسایه‌هایش بدهد هزینه رسیدن به آنهایی را که قطعاً باید از همان مسیریاب بگذرند را اعلام نمی‌کند. (یا ∞ اعلام می‌کنند)

مسئله شمارش تا بینهایت

به خبرهای خوب واکنش سریع ولی به خبرهای بد واکنش کندی نشان می دهد.

A	B	C	D	E	
●	●	●	●	●	
	∞	∞	∞	∞	Initially
	1	∞	∞	∞	After 1 exchange
	1	2	∞	∞	After 2 exchanges
	1	2	3	∞	After 3 exchanges
	1	2	3	4	After 4 exchanges

(a)

The count-to-infinity problem.

هرگاه مسیریابی از زیر شبکه خارج شود هر کدام از سایر مسیریاب‌های فعال احساس می‌کنند از طریق دیگری مسیری بهتر به آن وجود دارد.

A	B	C	D	E	
●	●	●	●	●	
	1	2	3	4	Initially
	3	2	3	4	After 1 exchange
	3	4	3	4	After 2 exchanges
	5	4	5	4	After 3 exchanges
	5	6	5	6	After 4 exchanges
	7	6	7	6	After 5 exchanges
	7	8	7	8	After 6 exchanges
		⋮			
	∞	∞	∞	∞	

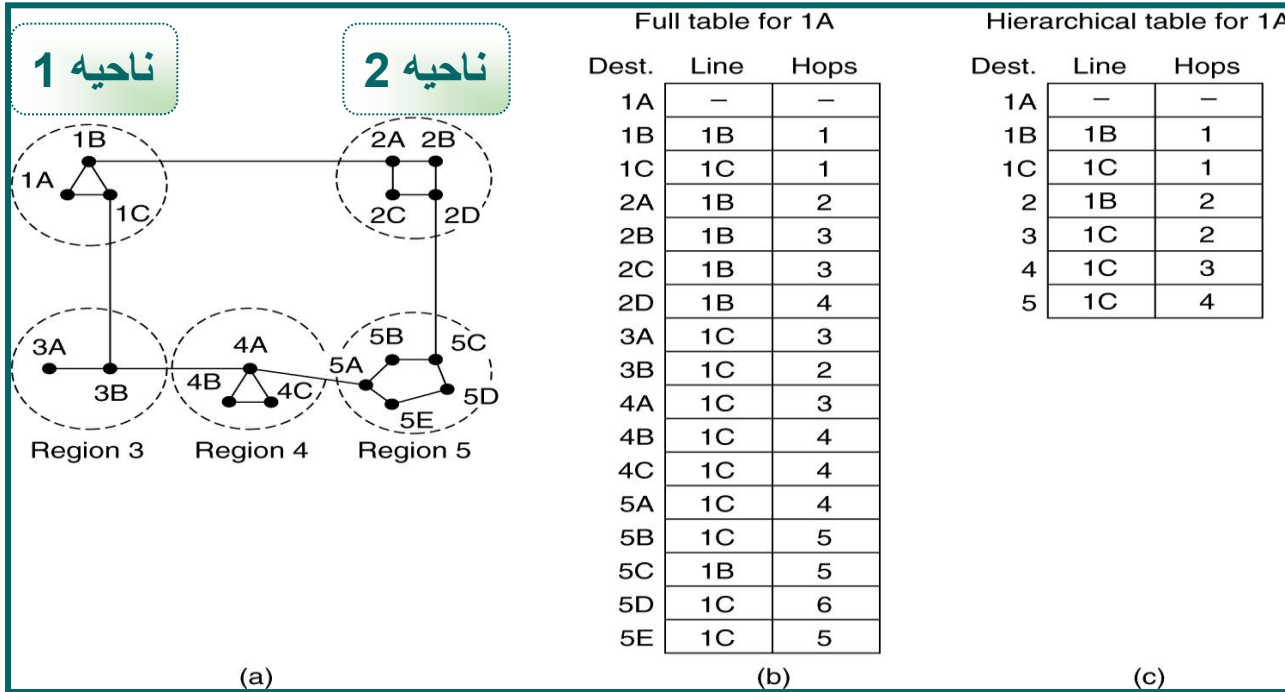
(b)

مسیریابی سلسله‌مراتبی Hierarchical Routing

رشد شبکه و زیاد شدن شبکه‌های محلی و مسیریابها، افزایش حجم جداول مسیریابی و زیاد شدن زمان لازم جهت تعیین مسیر یک بسته و در نتیجه ایجاد تأخیرهای بحرانی و کاهش کارایی شبکه

در مسیریابی سلسله‌مراتبی، مسیریابها در گروههایی به نام "ناحیه **Region**" دسته‌بندی می‌شوند. هر مسیریاب فقط "نواحی" و مسیریابهای درون ناحیه خود را می‌شناسد و هیچ اطلاعی از مسیریابهای درون نواحی دیگر ندارد.

مسیریابی سلسله مراتبی



مشکل روش سلسله مراتبی

به دلیل مشخص نبودن کل توپولوژی زیرشبکه برای هر مسیر یاب :
ممکن است مسیر انتخابی جهت ارسال بسته به یک مسیر یاب خاص درون یک ناحیه بهینه نباشد.

مزیت استفاده از روشهای سلسله مراتبی: صرفه جویی در اندازه جداول مسیریابی

	تعداد ناحیه Regions	تعداد دسته Clusters	تعداد حوزه Zones	تعداد مسیر یاب	تعداد رکورد در جدول
مسیریابی DV بدون سلسله مراتب	۱	—	—	۷۲۰	۷۲۰
مسیریابی DV با سلسله مراتب دو سطحی	۲۴	—	—	۳۰	۵۳
مسیریابی DV با سلسله مراتب سه سطحی	۹	۸	—	۱۰	۲۵
مسیریابی DV با سلسله مراتب سه سطحی	۹	۵	۴	۴	۱۹

مقایسه اندازه جدول مسیریابی در روشهای سلسله مراتبی

اینترنت مجموعه‌ای از شبکه‌های خودمختار **Autonomous** و “مستقل” است که به نحوی به هم متصل شده‌اند. شبکه خودمختار که اختصاراً **AS** نامیده می‌شود، شبکه‌ای است که تحت نظارت و سرپرستی یک مجموعه یا سازمان خاص پیاده و اداره می‌شود. مثلاً یک دانشگاه

مسئول شبکه خودمختار می‌تواند بر روی شبکه تحت نظارت خود “حاکمیت” داشته باشد یعنی ، ، سیستم عامل می‌تواند بر روی تک تک اجزای شبکه (ماشینهای میزبان)، توپولوژی کل شبکه طراحی زیرساخت ارتباطی و طریقه اتصال شبکه‌های محلی و نوع پروتکل مسیریابی اعمال نفوذ کرده و نظرات خود را پیاده نماید.

مسیریابی در شبکه های خود مختار

مسیریابی بسته های IP در درون یک شبکه خودمختار بیشتر تابع پارامترهایی نظیر سرعت و قابل اعتماد بودن الگوریتم مسیریابی است .

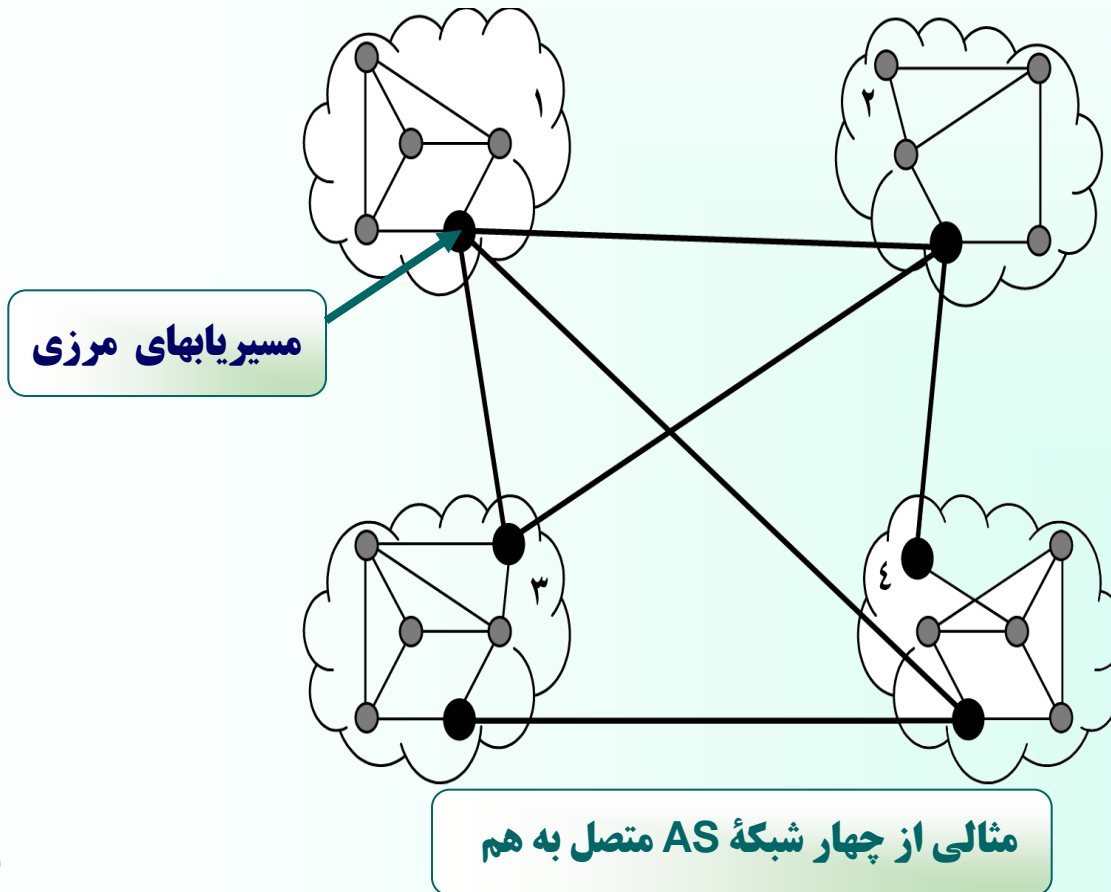
دروازه های مرزی Border Gateway :

مسیریابهایی که ارتباط دو شبکه خودمختار متفاوت را برقرار می کنند و تمامی ارتباطات بین شبکه ای از طریق آنها انجام می شود .

دروازه های مرزی Interior Gateway
مسیریابهایی که ارتباط دو شبکه خودمختار متفاوت را برقرار می کنند و تمامی ارتباطات بین شبکه ای از طریق آنها انجام می شود.

- مسیریابهای مرزی و ساختار ارتباطی بین آنها تابع قواعد "مسیریابی برونی"
- مسیریابهای داخلی تابع الگوریتمهای "مسیریابی درونی" مرزی
- مسیریابهای مرزی = مسیریابهای BGP

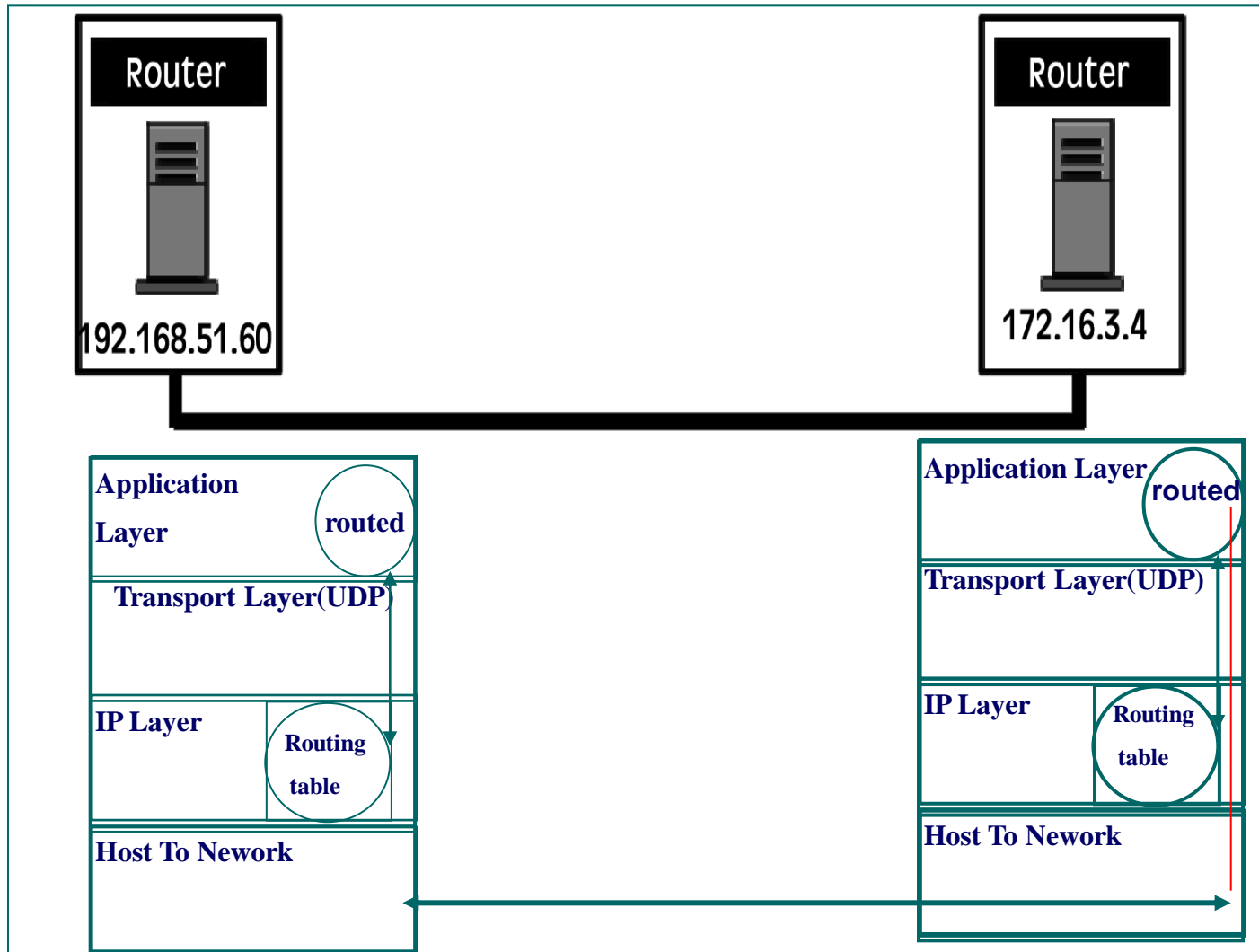
- مثال:** اگر یک ماشین میزبان در شبکه ۱ بخواهد بسته‌ای برای ماشین دیگر در شبکه ۴ بفرستد سه مرحله مسیریابی لازم است:
- مسیریابی در درون شبکه ۱ تا رسیدن بسته به مسیریاب مرزی
 - مسیریابی روی خطوط ارتباطی بین شبکه‌های ۱ تا رسیدن به شبکه ۴
 - مسیریابی درون شبکه ۴ تا رسیدن به ماشین مقصد



پروتکل RIP در مسیریابی درونی : Routing Information Protocol

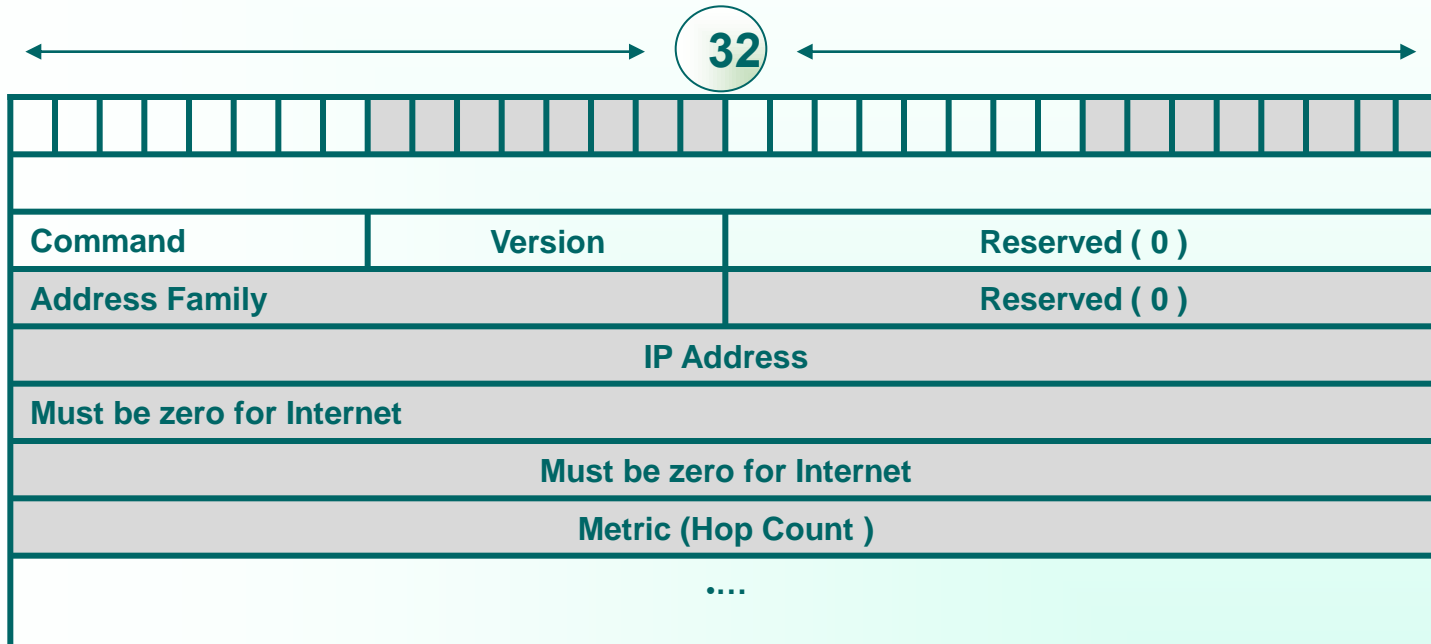
- اولین پروتکل مسیریابی درونی (1982)
- مبتنی بر الگوریتم بردار فاصله DV
- معیار هزینه = تعداد گام
- مبادله جداول مسیریابی هر ۳۰ ثانیه یکبار بین مسیریابهای مجاور
- حداکثر تعداد طول مسیر = ۱۵
- استفاده از پروتکل UDP و پورت شماره ۲۵۰ جهت مبادله جداول مسیریابی

جداول مسیریابی در لایه دوم جهت مسیریابی بسته‌های IP
مبادله جداول و عملیات به هنگام‌سازی توسط برنامه کاربردی لایه چهارم



پروتکل RIP در لایه کاربرد

قالب پیامها در پروتکل RIP



پروتکل OSPF در مسیریابی درونی Open Shortest Path First

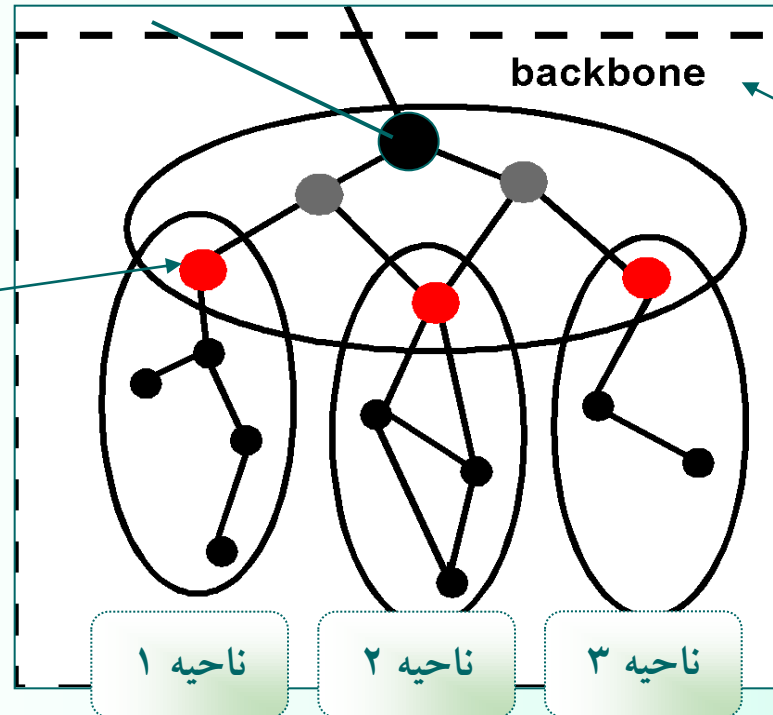
مقایسه پروتکل OSPF با RIP

- استفاده از الگوریتم **LS** برای محاسبه بهترین مسیر بر خلاف پروتکل **RIP** و عدم وجود مشکل "شمارش تا بینهایت"
- توانایی در نظر گرفتن چندین معیار هزینه در انتخاب بهترین مسیر برخلاف پروتکل **RIP**
- در نظر گرفتن حجم بار و ترافیک یک مسیریاب در محاسبه بهترین مسیر بر خلاف پروتکل **RIP** و همگرایی سریع جداول مسیریابی در هنگام خرابی یک مسیریاب
- انتخاب مسیر مناسب برای یک بسته بر اساس نوع سرویس درخواستی با توجه به فیلد **Type of Service** در بسته **IP** بر خلاف پروتکل **RIP**

مقایسه پروتکل OSPF با RIP

- هدایت نکردن تمام بسته‌های ارسالی برای یک مقصد خاص، روی بهترین مسیر و ارسال درصدی از بسته‌ها روی مسیرهای در رتبه ۲ و ۳ و ... از نظر هزینه، برخلاف پروتکل RIP = موازنه = **Load Balancing**
- پشتیبانی از مسیریابی سلسله‌مراتبی برخلاف پروتکل RIP
- عدم قبول جداول مسیریابی مسیریابها توسط هر مسیریاب بدون احراز هویت ارسال‌کننده آن
- استفاده مستقیم از پروتکل IP برخلاف پروتکل RIP (استفاده از پروتکل UDP در لایه انتقال)

- تقسیم یک شبکه خود مختار به تعدادی ناحیه و اطلاع تمام مسیریابهای درون یک ناحیه از مسیریابهای هم ناحیه و هزینه ارتباط بین آنها و ذخیره آن در جدول
- ارسال جداول برای تمام مسیریابهای هم ناحیه در زمانهای بهنگام سازی



مسیریابهای مرزی
برقرار کننده ارتباط نواحی

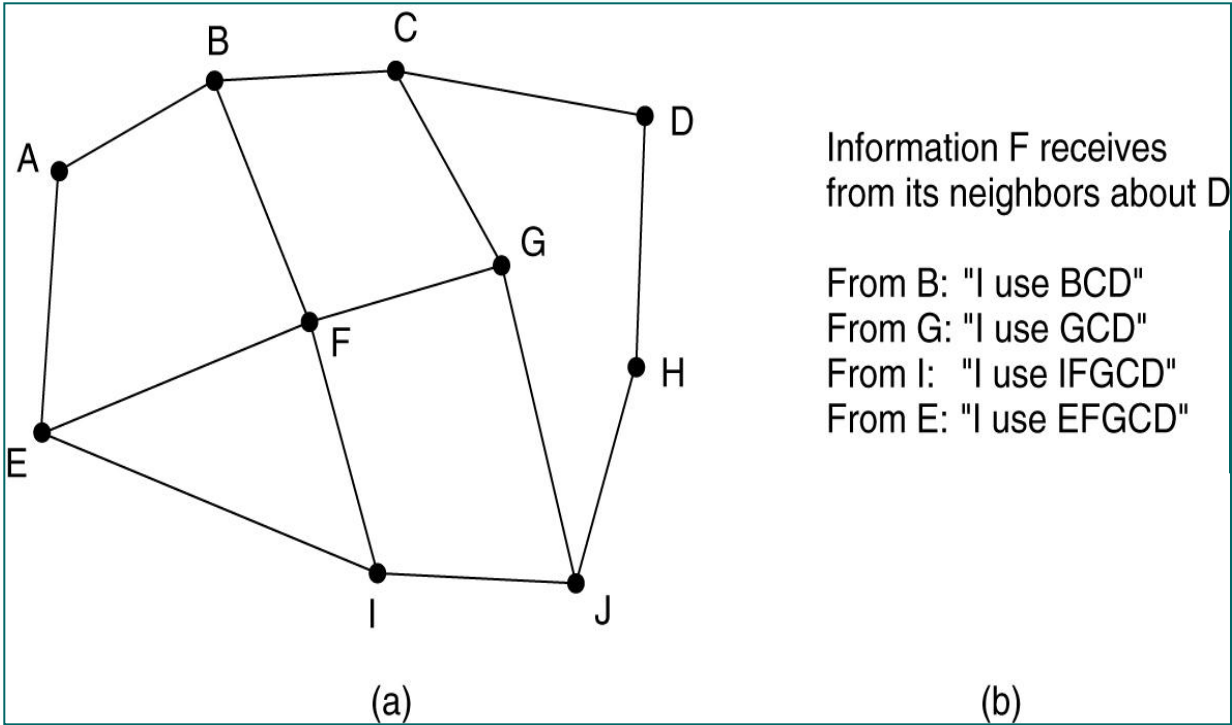
مجموعه مسیریابهای مرزی +
سیریابهای خارج از هر ناحیه +
ساختار ارتباطی بین این مسیریابها

سلسله مراتب مسیریابی در پروتکل OSPF

پروتکل BGP : پروتکل مسیریابی برون‌ی The Exterior Gateway Routing Protocol

- الگوریتمهای مسیریابی بین شبکه‌های خود مختار در اینترنت : **BGP**
- به جای مبادله جداول مسیریابی و هزینه‌ها در پروتکل **BGP** بین مسیریابهای مجاور، ارسال فهرستی از مسیرهای کامل بین هر دو مسیریاب در شبکه برای مسیریابهای مجاور در بازه‌های زمانی **T** ثانیه‌ای (بدون تعیین هزینه)

دریافت اطلاعات توسط مسیریاب F در مورد مسیریاب D از مسیریابهای مجاور



- تعیین مسیر رسیده از B
- تعیین مسیر رسیده از G
- تعیین مسیر رسیده از I
- تعیین مسیر رسیده از E

BGP ساختار فرضی از ارتباط بین مسیریابهای

الگوریتمهایی که در تبادل اطلاعات با همسایگان مسیرهای کامل را به اطلاع یکدیگر می‌رسانند:

اولاً: مشکل "شمارش تا بینهایت" را نخواهد داشت. مانند پروتکل **BGP**

ثانیاً: مسیریابهای دیگر می‌توانند بر روی کل مسیر، بررسی‌های امنیتی، اقتصادی، سیاسی و ملی انجام دهند و بر اساس این پارامترها مسیر مناسب را انتخاب نمایند. مانند پروتکل **BGP**

تبادل اطلاعات مسیریابی (فهرست مسیرها) در پروتکل **BGP** در قالب پیام

انواع پیام تعریف شده در پروتکل **BGP**:

1. پیام **OPEN**
2. پیام **KEEPALIVE**
3. پیام **NOTIFICATION**
4. پیام **UPDATE**

فصل پنجم : لایه انتقال در شبکه اینترنت

هدفهای آموزشی :

- مفاهیم لایه انتقال
- مفهوم پورت و سوکت
- تشریح پروتکل TCP
- روش برقراری ارتباط در پروتکل TCP
- روش کنترل جریان داده‌ها در پروتکل TCP
- زمان سنجها و عملکرد آنها در پروتکل TCP
- پروتکل UDP



پروتکل‌های لایه انتقال

UDP

User Datagram
Protocol

TCP

Transmission Control
Protocol

لایه IP

- هدایت و مسیریابی بسته‌های اطلاعاتی از یک ماشین میزبان به ماشین دیگر
- عدم حل مشکلات احتمالی به وجود آمده برای بسته‌های IP در مسیر

لایه انتقال

- فراهم آوردن خدمات سازماندهی‌شده، مبتنی بر اصول سیستم عامل، برای برنامه‌های کاربردی در لایه بالاتر
- جبران کاستی‌های لایه IP

راهکارهای پروتکل TCP

- برقراری یک ارتباط و اقدام به هماهنگی بین مبدأ و مقصد قبل از ارسال هر گونه داده

- قراردادن شماره ترتیب برای داده‌ها
- تنظیم کد ۱۶ بیتی کشف خطا در مبدأ و بررسی مجدد آن در مقصد جهت اطمینان از صحت داده‌ها

کاستی‌های لایه IP

- عدم تضمین درآماده‌بودن ماشین مقصد جهت دریافت بسته

- عدم تضمین در به ترتیب رسیدن بسته‌های متوالی و داده‌ها و صحت آنها

راهکارهای پروتکل TCP

کاستی‌های لایه IP

❖ قرار دادن شماره ترتیب در بسته ارسالی

❖ عدم تمایز در دریافت بسته‌های تکراری در مقصد
(Duplication Problem)

➤ استفاده از الگوریتم پویا جهت تنظیم مجموعه زمانسجها

➤ عدم تنظیم سرعت ارسال و تحویل بسته‌ها

□ قراردادن آدرس پورت پروسه فرستنده و گیرنده در سرآیند بسته ارسالی

□ عدم توزیع بسته‌ها بین پروسه‌های مختلف اجرا شده بر روی یک ماشین واحد

آدرس پورت

شماره شناسایی مشخص کننده هر پروسه برای برقراری یک ارتباط با پروسه‌ی دیگر بر روی شبکه

شماره پورتهای استاندارد

Port	Protocol	Use
21	FTP	File transfer
23	Telnet	Remote login
25	SMTP	E-mail
69	TFTP	Trivial File Transfer Protocol
79	Finger	Lookup info about a user
80	HTTP	World Wide Web
110	POP-3	Remote e-mail access
119	NNTP	USENET news

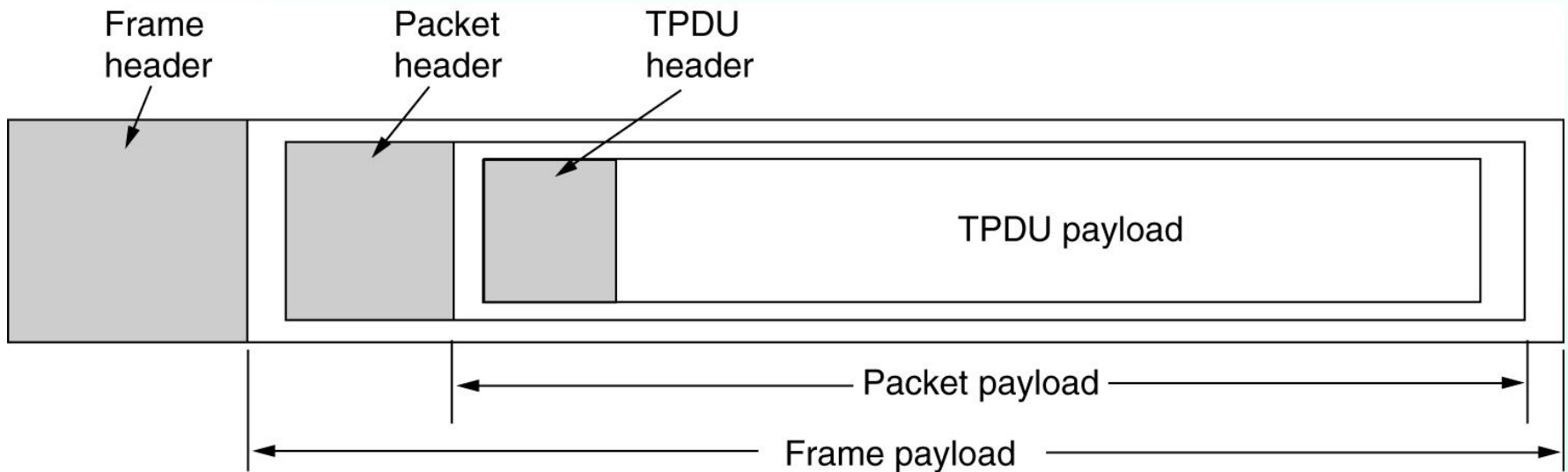
آدرس سوکت

زوج آدرس IP و آدرس پورت مشخص کننده یک پروسه یکتا و واحد بر روی هر ماشین در دنیا

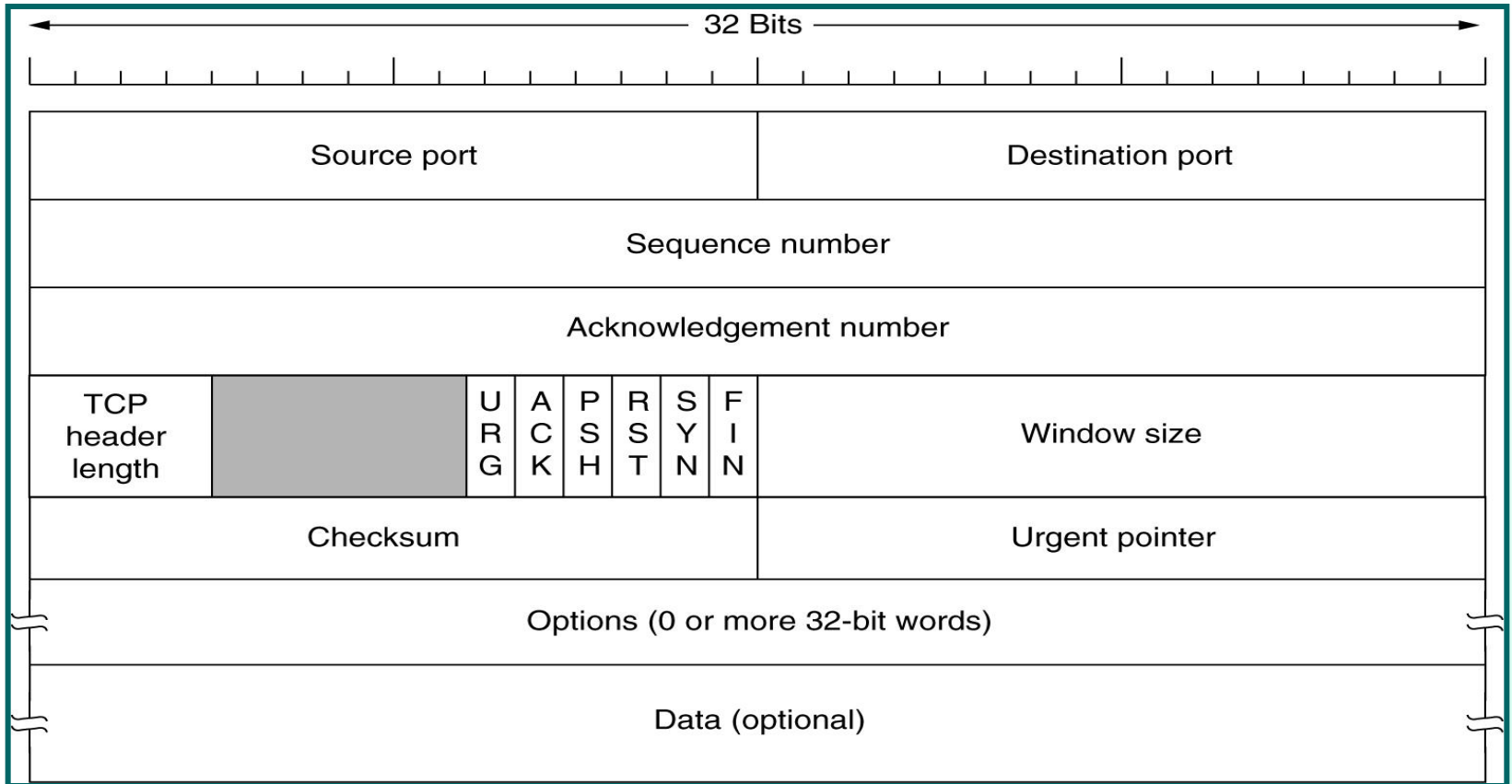
(IP Address: Port Number) = Socket Address

مثال 80 : ۱۹۳.۱۴۲.۲۲.۱۲۱

TCP = Transport Protocol Data Unit = بسته تولید شده در لایه انتقال = قطعه TCP



بسته پروتکل TCP



فیلد Source Port

- فیلد ۱۶ بیتی
- آدرس پورت پروسه مبدأ

فیلد Destination Port

- فیلد ۱۶ بیتی
- آدرس پورت پروسه مقصد

فیلد Sequence Number

- فیلد ۳۲ بیتی
- مشخص کننده شماره ترتیب آخرین بایت قرار گرفته شده در فیلد داده از بسته جاری

فیلد Acknowledgement Number

- فیلد ۳۲ بیتی
- مشخص کننده شماره ترتیب بایتی که فرستنده بسته منتظر دریافت آن است

فیلد TCP Header Length

- فیلد ۴ بیتی
- مشخص کننده طول سرآیند بسته **TCP** بر مبنای کلمات ۳۲ بیتی
- حداقل مقدار = ۵
- تعیین کننده محل شروع داده‌ها در بسته **TCP**

۶ بیت بلا استفاده

۶ بیت بلا استفاده جهت استفاده در آینده

بیت‌های Flag

۶ بیتی

U	A	P	R	S	F
R	C	S	S	Y	I
G	K	H	T	N	N

بیت URG

مقدار فیلد = ۱ نشان دهنده معتبر بودن مقدار موجود در فیلد **Urgent Pointer**

مقدار فیلد = ۰ نشان دهنده نا معتبر بودن مقدار موجود در فیلد **Urgent Pointer**

U	A	P	R	S	F
R	C	S	S	Y	I
G	K	H	T	N	N

بیت ACK

مقدار فیلد = ۱ نشان‌دهنده معتبر بودن مقدار موجود در فیلد
Acknowledgement Number

بیت PSH (PUSH)

مقدار فیلد = ۱ ← نشان‌دهنده تقاضای فرستنده اطلاعات از گیرنده
اطلاعات جهت بافرنکردن داده‌های موجود در بسته و تحویل سریع بسته به
برنامه‌های کاربردی به منظور انجام پردازشهای بعدی

بیت RST

مقدار فیلد = ۱ ← نشان‌دهنده قطع ارتباط به صورت یکطرفه و ناهماهنگ

بیت SYN

تغییر مقدار این فیلد جهت برقراری ارتباط توسط ماشین

روند برقراری ارتباط TCP

الف) تنظیم بیت‌های $ACK=0$ و $SYN=1$ توسط شروع کننده ارتباط در یک بسته TCP بدون داده (تقاضای برقراری ارتباط = Connection Request)

ب) تنظیم بیت‌های $SYN=1$ و $ACK=1$ در صورت قبول طرف دریافت کننده بسته تقاضای برقراری ارتباط به برقراری ارتباط

بیت FIN

مشخص کننده قطع و پایان ارسال اطلاعات هنگام اتمام داده‌های ارسالی توسط طرفین با ۱ نمودن مقدار این بیت هنگام ارسال آخرین بسته

قطع کامل ارتباط: ۱ نمودن مقدار این فیلد توسط هر دو ماشین فرستنده و گیرنده

قطع ارتباط یکطرفه: ۱ نمودن مقدار این فیلد توسط یکی از طرفین ارتباط

فیلد Windows Size

مشخص کننده مقدار ظرفیت خالی فضای بافر گیرنده

فیلد Checksum

- فیلد ۱۶ بیتی
- حاوی کد کشف خطا

طریقه محاسبه کد کشف خطا

- تقسیم کل بسته TCP به قالبهای ۱۶ بیتی (منهای قسمت Checksum)
- ایجاد یک سرآیند فرضی و تقسیم آن به صورت کلمات ۱۶ بیتی
- جمع تمامی کلمات در مبنای مکمل ۱ و منفی نمودن عدد حاصل در مبنای مکمل ۱ و قرارگرفتن عدد حاصل در فیلد Checksum

جمع کل کلمات ۱۶ بیتی موجود در بسته TCP + سرآیند فرضی = ۰ ← عدم بروز خطا در حین ارسال داده‌ها

ساختار سرآیند فرضی

- ۳۲ بیت آدرس IP ماشین مبدأ
- ۳۲ بیت آدرس IP ماشین مقصد
- یک فیلد ۸ بیتی کاملاً صفر
- فیلد ۸ بیتی پروتکل که برای پروتکل TCP = ۶
- فیلد TCP Segment Length = طول کل بسته TCP

Source IP Address																															
Destination IP Address																															
00000000								00000110								TCP Segment Length															

فیلد Urgent Pointer

اشاره گر به موقعیت داده‌های اضطراری موجود
در بسته TCP

فیلد Option

- فیلد اختیاری
- شامل مقدار حداکثر طول بسته
- قراردادن کدهای بی ارزش در این فیلد به جهت آنکه طول بسته ضریبی از ۴ باقی بماند

روش برقراری ارتباط در پروتکل TCP

روش دست تکانی سه مرحله‌ای

مرحله اول:

- ارسال یک بسته TCP خالی از داده از طرف شروع‌کننده ارتباط با بیتیهای $SYN=1$ و $ACK=0$ و قراردادن عدد X درون فیلد شماره ترتیب

- اعلام شروع ترتیب داده‌های ارسالی از $X+1$ به ماشین طرف مقابل
- پیشگیری از مساوی بودن شماره ترتیب داده‌های ارسالی با انتخاب مقدار X به صورت تصادفی

مرحله دوم:

- رد تقاضای برقراری ارتباط: ارسال بسته‌ای خالی با بیت $RST=1$
- قبول تقاضای برقراری ارتباط: ارسال بسته خالی با مشخصات زیر از طرف گیرنده بسته تقاضا:

• بیت $SYN = 1$

• بیت $ACK = 1$

• $Acknowledgement = x+1$

• $Sequence Number = y$

روش دست تکانی سه مرحله‌ای

مرحله سوم:

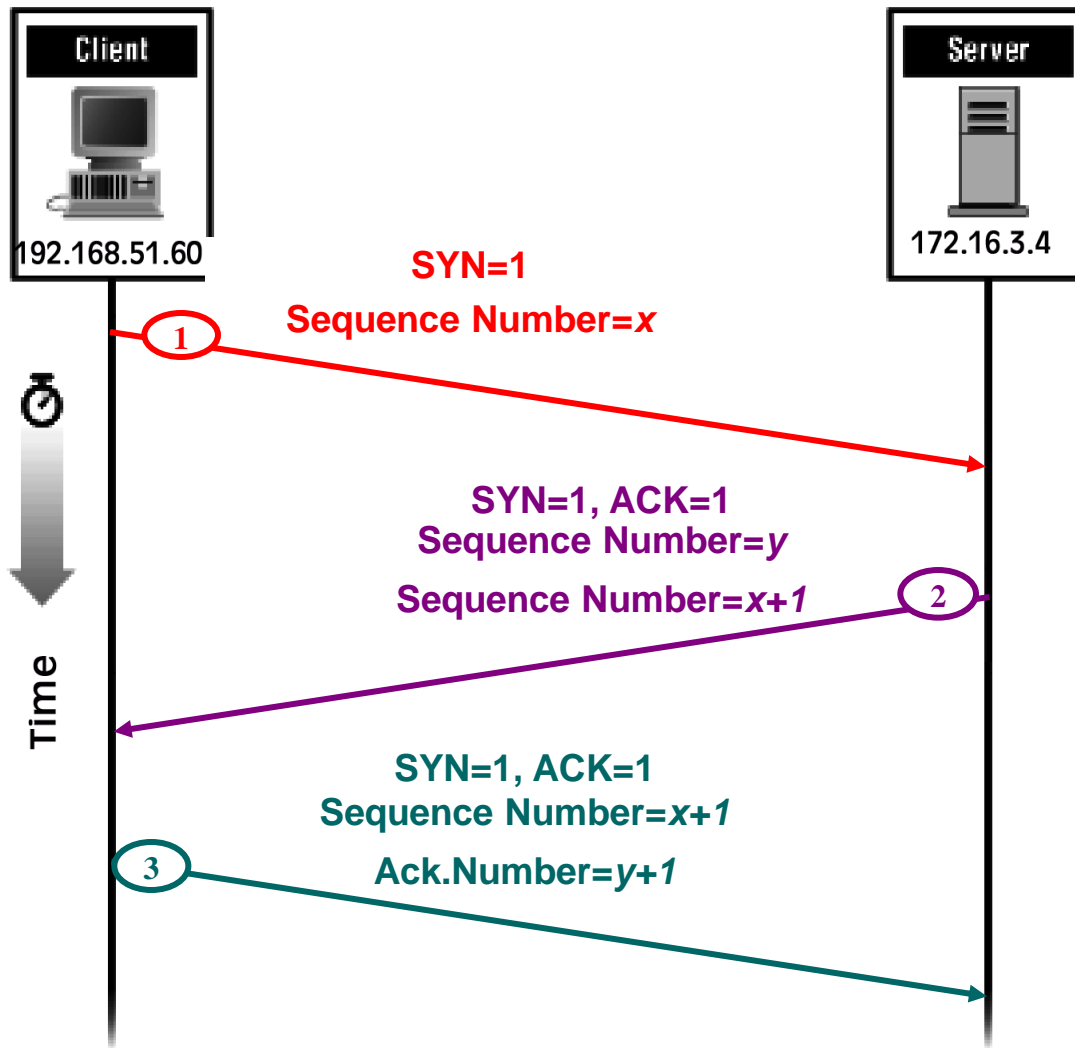
تصدیق شروع ارتباط از طرف شروع‌کننده ارتباط با قراردادن مقادیر زیر در بیت‌های:

$$\text{SYN} = 1 \bullet$$

$$\text{ACK} = 1 \bullet$$

$$\text{Acknowledgement Number} = y + 1 \bullet$$

$$\text{Seq. No} = x + 1 \bullet$$



مراحل دست تکانی سه مرحله ای برای برقراری ارتباط در پروتکل TCP

روند خاتمه ارتباط TCP

- ارسال بسته TCP با بیت $FIN = 1$ از طرف درخواست کننده اتمام ارسال
- موافقت طرف مقابل با اتمام ارتباط یکطرفه و ادامه ارسال داده توسط آن
- قطع ارتباط دو طرفه با یک نمودن مقدار بیت FIN در آخرین بسته ارسالی و تصدیق پایان ارتباط از طرف مقابل

کنترل جریان در پروتکل TCP

- استفاده از بافر جهت کنترل جریان داده‌ها در پروتکل TCP
- بافر شدن داده‌ها قبل از ارسال به برنامه کاربردی لایه بالاتر
- امکان عدم دریافت و ذخیره داده‌ها توسط برنامه کاربردی در مهلت مقرر و پر شدن بافر
- ← اعلام حجم فضای آزاد بافر در فیلد **Window** در هنگام ارسال بسته TCP به طرف مقابل
- ایجاد یک ساختمان داده خاص به ازای هر ارتباط برقرار شده TCP و نگهداری اطلاعاتی از آخرین وضعیت ارسال و دریافت جریان داده‌ها = ساختمان داده بلوک نظارت بر انتقال = **TCB = Transmission Control Block**

نام متغیر	توضیح
متغیرهای نظارت بر ارسال داده‌ها	
SND.UNA	شماره ترتیب آخرین بسته ای که ارسال شده ولی هنوز پیغام Ack آن برنگشته است.
SND.NXT	شماره ترتیب آخرین بایت که داده ها از آن شماره به بعد در بسته بعدی که باید ارسال شود.
SND.WND	میزان فضای آزاد در بافر ارسال
SND.UP	شماره ترتیب آخرین داده های اضطراری که تحویل برنامه کاربردی شده است.
SND.WL1	
SND.WL2	
SND.PUSH	شماره ترتیب آخرین داده هایی که باید آنی به برنامه کاربردی گسیل (Push) شود.
SND.ISS	مقدار اولیه شمارنده ترتیب داده های دریافتی که در حین ارتباط بر روی آن توافق می شود.
متغیرهای نظارت بر دریافت داده‌ها	
RCV.NXT	شماره ترتیب آخرین بایت در بسته بعدی که از آن شماره به بعد انتظار دریافت آنرا دارد.
RCV.WND	میزان فضای آزاد در بافر دریافت
RCV.UP	شماره ترتیب آخرین داده های اضطراری که برای برنامه طرف مقابل ارسال شده است.
RCV.IRS	مقدار اولیه شمارنده ترتیب داده های ارسالی که در حین ارتباط بر روی آن توافق می شود.

متغیرهای ساختمان داده TCP

فرستنده



گیرنده



فضای بافرگیرنده



4 Kbyte

ارسال 2 Kbyte داده



Window Size=2048

ارسال 2 Kbyte داده



Window Size=0

فرستنده متوقف می‌شود

گیرنده 2KB از بافر میخواند



Window Size=2048

فرستنده مجدداً احیا می‌شود

ارسال 1 Kbyte داده



مثال روند کنترل جریان در پروتکل TCP

زمان سنجها در پروتکل TCP

TCP Timer

وابستگی عملکرد صحیح پروتکل TCP به استفاده درست از
زمان سنجها

زمان سنجها

Retransmission Timer

Keep-Alive Timer

Persistence Timer

Quite Timer

Idle Timer

زمان سنج Retransmission Timer

پس از برقراری ارتباط و ارسال بسته برای پروسه مقصد، زمان‌سنجی (RT) با مقدار پیش فرض تنظیم و فعال می‌گردد و شروع به شمارش معکوس می‌نماید که اگر در مهلت مقرر پیغام دریافت بسته (Ack) نرسید رخداد انقضای زمان تکرار روی داده و ارسال مجدد بسته صورت گیرد.

**Retransmission
Timeout Event**

عملکرد این زمان سنج **Retransmission Timer** بسیار ساده است اما مشکل در اینجاست که:

- ۱- عمل ارسال مجدد یک بسته چند بار باید تکرار شود؟
- ۲- مقدار پیش فرض زمان سنج چه مقدار باشد؟

بهترین راه تنظیم زمان سنج: روشهای **وفقی و پویا**

الف) ایجاد یک متغیر حافظه به نام RTT و مقداردهی آن هنگام برقراری یک ارتباط TCP

ب) تنظیم یک زمان سنج به ازای ارسال هر بسته و اندازه زمان رفت و برگشت پیغام دریافت بسته $M =$

الگوریتم **Jacobson**

ج) بهنگام شدن مقدار پیش فرض زمان سنج از رابطه:

$$RTT_{new} = RTT_{old} + 4 * D_{new}$$
$$D_{new} = \alpha * D_{old} + (1 - \alpha) * (RTT_{old} - M)$$

$$\alpha = 7/8$$

مقدار اولیه D می تواند صفر باشد.

Keep-Alive Timer

- توقف ارسال اطلاعات و عدم تبادل داده علی رغم فعال و باز بودن ارتباط **TCP**
- قطع ارتباط یکی از طرفین به دلیل خرابی سخت افزاری و یا نرم افزاری

جهت تمایز این دو حالت

ارسال بسته **TCP** خالی از داده از طرف فرستنده اطلاعات برای مقصد با استفاده از زمان سنج **Keep-Alive Timer** (زمان پیش فرض بین ۵ تا ۴۵ ثانیه)

عدم بازگشت پیغام دریافت

قطع ارتباط به صورت یکطرفه و آزاد نمودن تمام بافرها

بازگشت پیغام دریافت از طرف مقصد

ارتباط **TCP** باز و فعال است

- مقدار فضای بافر آزاد یکی از طرفین ارتباط صفر (**Window Size= 0**) ← متوقف شدن پروسه طرف مقابل
- خالی شدن مقداری از فضای بافر پر شده بعد از مدتی ← اعلام آزاد شدن فضای بافر جهت احیای پروسه بلوکه و متوقف شده توسط سیستم عامل و شروع و ادامه ارسال پروسه متوقف شده

Persistence Timer

ارسال بسته TCP در فواصل زمانی منظم با استفاده از زمان سنج Persistence Timer پس از آزاد شدن فضای بافر برای پروسه بلوکه شده جهت احیا و ادامه ارسال داده توسط آن

Quite Timer

هنگام بسته شدن یک ارتباط **TCP** با شماره پورت خاص تا مدت زمان معینی که زمان سنج **Quite Timer** تعیین می نماید (مقدار پیش فرض = ۳۰ تا ۱۲۰ ثانیه) هیچ پروسه ای اجازه استفاده از شماره پورت بسته شده را ندارد.

جهت رسیدن بسته های سرگردان ناشی از ارتباط پایان یافته موجود در شبکه به مقصد

Idle Timer

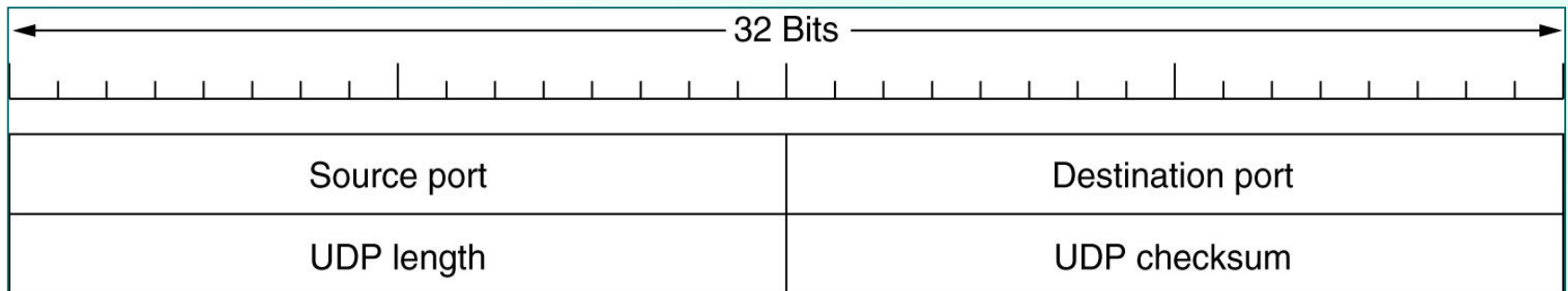
اگر تلاش برای تکرار ارسال یک بسته بیش از حد متعارف انجام شود ارتباط **TCP** را بصورت یکطرفه رها کرده و قطع می نماید. مقدار معمول آن ۳۶۰ ثانیه است.

پروتکل UDP

ارسال بسته به مقصد بدون اطمینان
از برقراری ارتباط و آماده بودن
ماشین مقصد

- پروتکل بدون اتصال (Connectionless)
- پروتکل ساده و سریع
- کاربرد در سیستم های DNS و TFTP

بسته UDP



فیلدهای بسته UDP

فیلد Source Port

- فیلد ۱۶ بیتی
- مشخص کننده آدرس پورت پروسه مبدأ

فیلد Detination Port

- فیلد ۱۶ بیتی
- مشخص کننده آدرس پورت پروسه مقصد

فیلد UDP Length

- فیلد ۱۶ بیتی
- طول بسته UDP بر حسب بایت (شامل سرآیند و داده‌ها)

فیلد UDP Checksum

- فیلد ۱۶ بیتی
- درج کد کشف خطا در این فیلد
- فیلد اختیاری (جهت ارسال دیجیتال صدا و تصویر مقدار تمام بیتها صفر)

مناسبتترین کاربرد پروتکل UDP = پروسه هایی که عملیات آنها مبتنی بریک تقاضا و یک پاسخ می باشد.

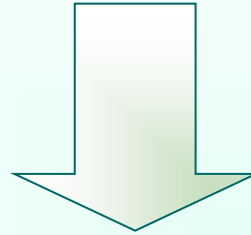
مانند : سیستم DNS

ماشینهای Little Edition و Big Edition

ماشینهای **Big Endian** : ماشینهایی که ابتدا بایت پر ارزش و سپس بایت کم ارزش را ذخیره می کنند مثل کامپیوترهای سری **SUN**

ماشینهای **little Endian** : ماشینهایی که ابتدا بایت کم ارزش و سپس بایت پر ارزش را ذخیره می کنند مثل کامپیوترهای شخصی با پردازنده سری **80X86** و پنتیوم

تشکیل بسته‌های IP ابتدا در حافظه و ارسال از طریق سخت افزار شبکه ← دریافت بسته IP
ارسالی از یک ماشین **Big Endian** به یک ماشین **Little Endian** و یا برعکس ←
تعویض بایتها و فاقد ارزش بودن محتوی بسته دریافتی



پروتکل **TCP/IP** ، استاندارد ماشین‌های **Big Endian** را مبنا قرار داده است

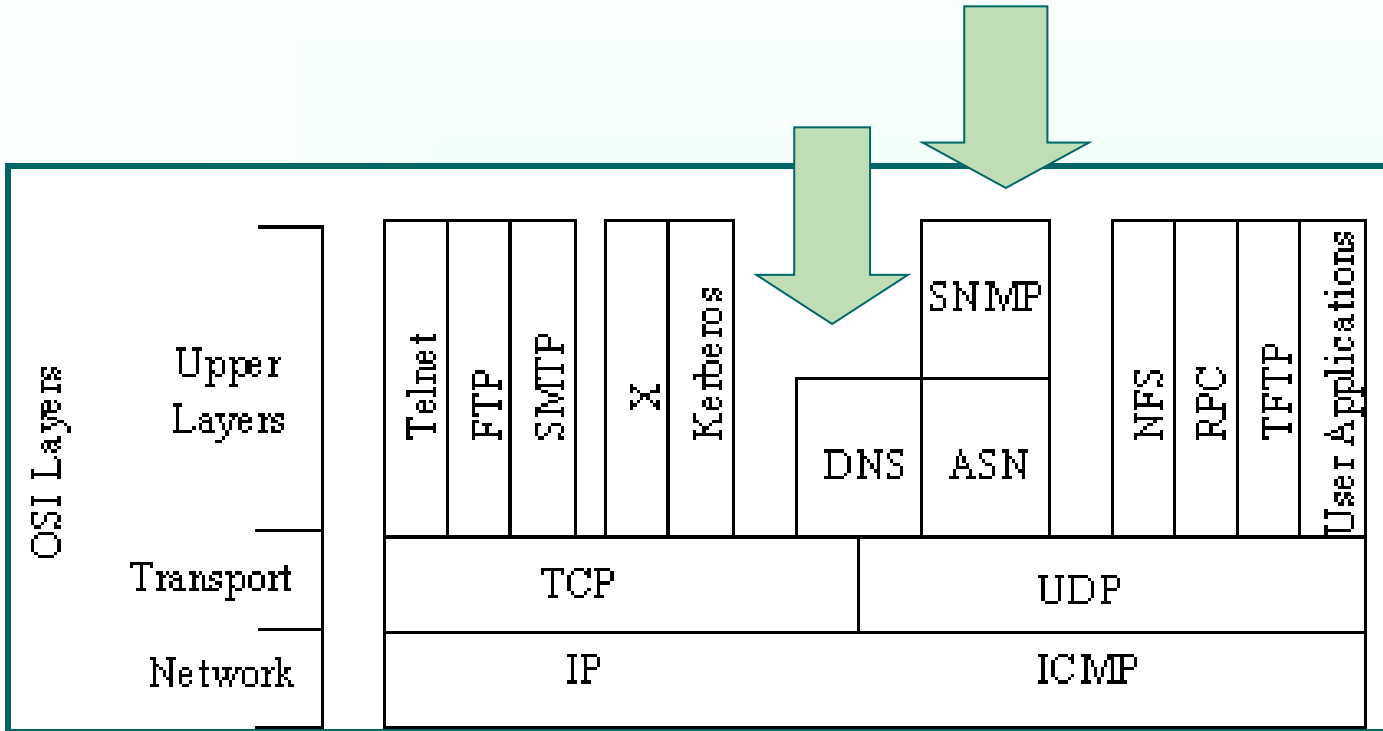
فصل ششم: سرویس دهنده‌های نام حوزه DNS و اصول مدیریت شبکه SNMP

هدفهای آموزشی:

- ❑ اصول سرویس دهنده‌های نام
- ❑ مفهوم نام حوزه و سلسله مراتب نام
- ❑ روشهای جستجو در سرویس دهنده‌های نام
 - ❖ پرس‌وجوی تکراری
 - ❖ پرس‌وجوی بازگشتی
 - ❖ پرس‌وجوی معکوس
- ❑ ساختار بانک اطلاعاتی در سرویس دهنده‌های نام
- ❑ قالب پیام در سرویس دهنده‌های نام حوزه
- ❑ اصول مدیریت شبکه در اینترنت
- ❑ اصول پروتکل **SNMP**



SNMP و DNS



سرویس دهنده نامهای حوزه (Domain Name System)

آدرسها در دنیای واقعی = آدرسهای اینترنت = آدرسهای نمادین = نام حوزه

مانند: www.ibm.com

ترجمه آدرسهای نمادین به آدرسهای IP

(۱) روش متمرکز: - تعریف تمام نامها و آدرسهای IP معادل در یک فایل به نام **hosts.txt**
- استفاده از فایل **hosts.txt** جهت ترجمه یک نام نمادین به آدرس IP معادل آن توسط تابع مترجم نام موجود در هر ماشین میزبان

کاربرد در شبکه ARPANET

و شبکه های کوچک و داخلی

۲) DNS یا سیستم نامگذاری حوزه:

- روشی سلسله مراتبی
- توزیع بانک اطلاعاتی مربوط به نامهای نمادین و معادل IP آنها در کل شبکه اینترنت
- معرفی این سیستم در سال ۱۹۸۴
- کاربرد در شبکه‌های بزرگ مانند اینترنت

روش ترجمه نام در DNS

- فراخوانی تابع تحلیلگر نام **Name Resolver** توسط برنامه کاربردی
- پارامتر ورودی تابع تحلیلگر نام آدرس نمادین
- ارسال بسته **UDP** (بسته درخواست) به آدرس یک سرویس‌دهنده **DNS** (به صورت پیش فرض مشخص می‌باشد) توسط تابع
- تحویل آدرس **IP** معادل با آدرس نمادین از طرف سرویس‌دهنده به تابع تحلیلگر
- تحویل آدرس **IP** به برنامه کاربردی درخواست‌کننده

نام حوزه

- تشکیل نام حوزه از بخشهایی به نام سطح
- تفکیک سطحها در نام حوزه با علامت •
- اشاره هر سطح از نام حوزه به یک قسمت از بانک اطلاعاتی توزیع شده
- تحلیل یک نام حوزه از سطوح سمت راست به چپ جهت پیدا نمودن سرویس‌دهنده متناظر

مثال :

www.yahoo.com

www.president.ir

هفت حوزه عمومی

.edu

موسسات علمی یا دانشگاهی
educational

.Com

موسسات اقتصادی و تجاری
commercial

.gov

آژانسهای دولتی آمریکا
government

.net

ارائه دهندگان خدمات شبکه
Network Service provider

.int

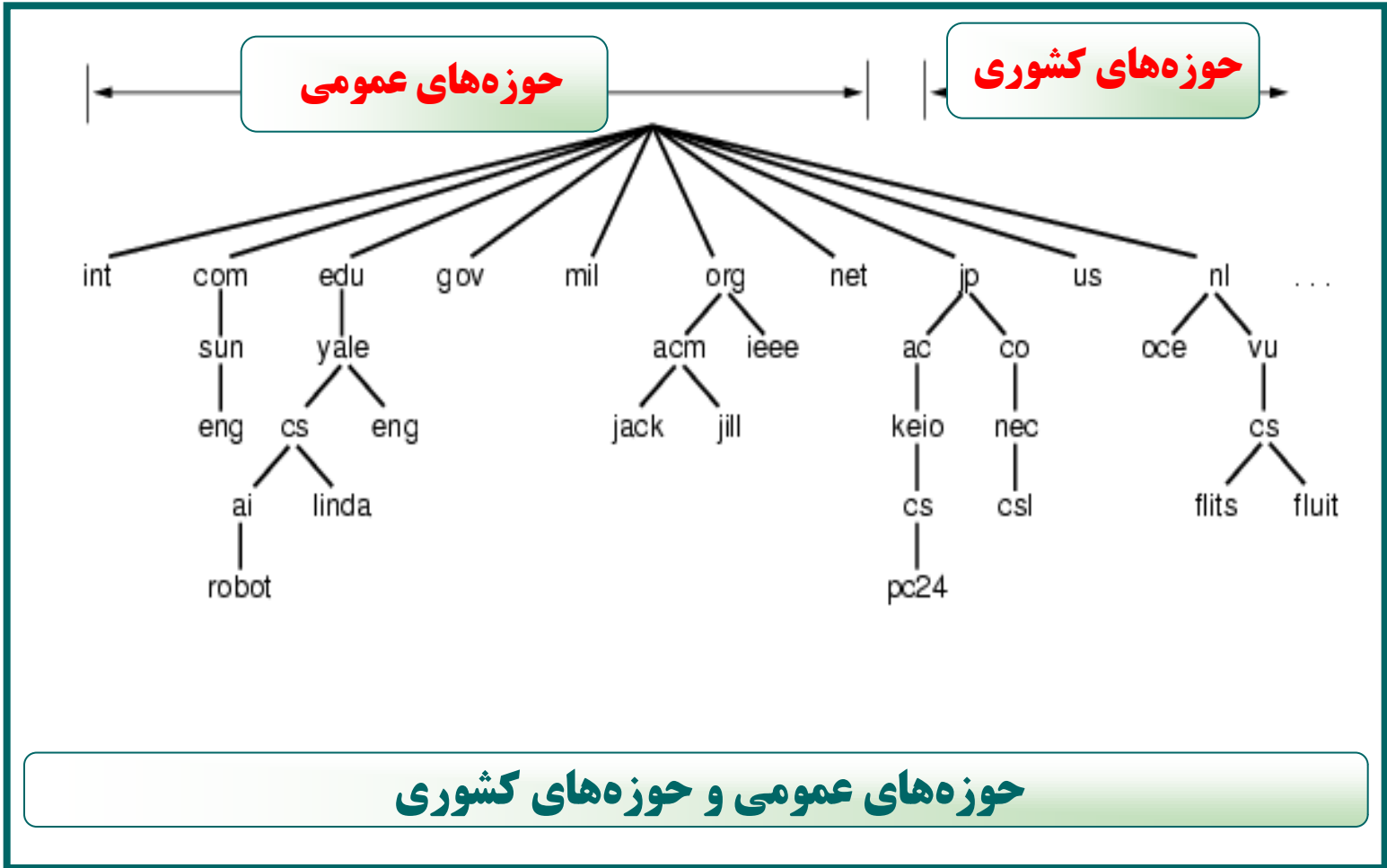
سازمانهای بین المللی
international

.org

سازمانهای غیر انتفاعی
organization

.mil

سازمانهای نظامی دنیا
military



روشهای جستجو در سرویس دهنده‌های نام

Iterative Query

• پرس و جوی تکراری

Recursive Query

• پرس و جوی بازگشتی

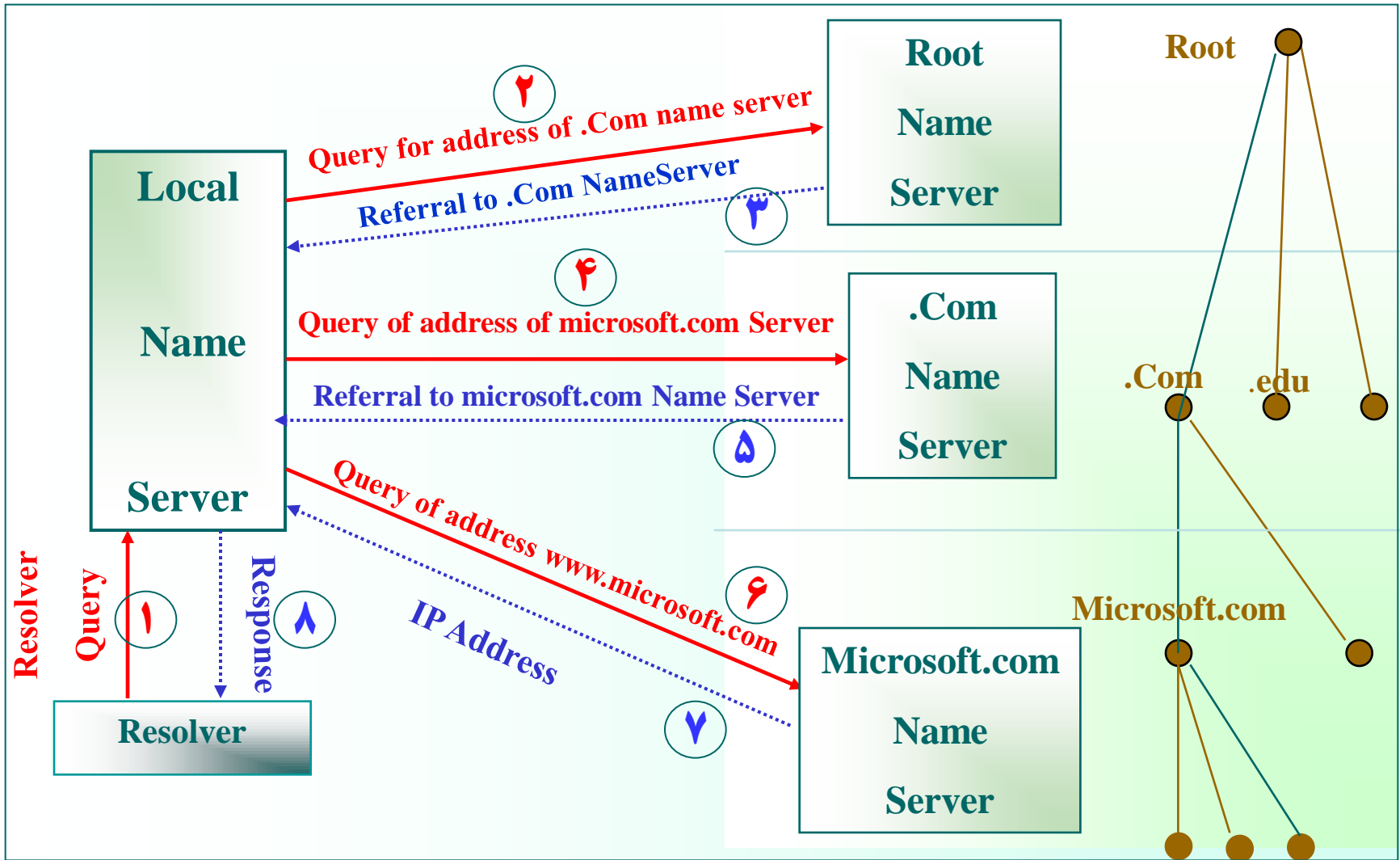
Reverse Query

• پرس و جوی معکوس

پرس و جوی تکراری

- حجم عمده عملیات بر عهده سرویس دهنده محلی
- داشتن آدرس ماشین **Root** به عنوان نقطه شروع توسط سرویس دهنده محلی
- ترجمه نام به آدرس **IP** بعد از دریافت تقاضای تبدیل نام توسط سرویس دهنده محلی و ارسال آن به تقاضا کننده در صورت امکان
- در غیر این صورت ارسال یک تقاضا برای **DNS** سطح بالا جهت ترجمه نام
- معرفی آدرس ماشین دیگر به سرویس دهنده محلی جهت ترجمه نام مورد نظر توسط سرویس دهنده سطح بالا
- ارسال تقاضا از طرف سرویس دهنده محلی به سرویس دهنده معرفی شده در مرحله قبل
- ترجمه نام حوزه توسط سرویس دهنده نام در غیر این صورت برگرداندن آدرس سرویس دهنده سطح پایین تر به سرویس دهنده محلی
- ادامه این روند تا ترجمه نام حوزه به آدرس **IP** توسط **DNS** نهایی

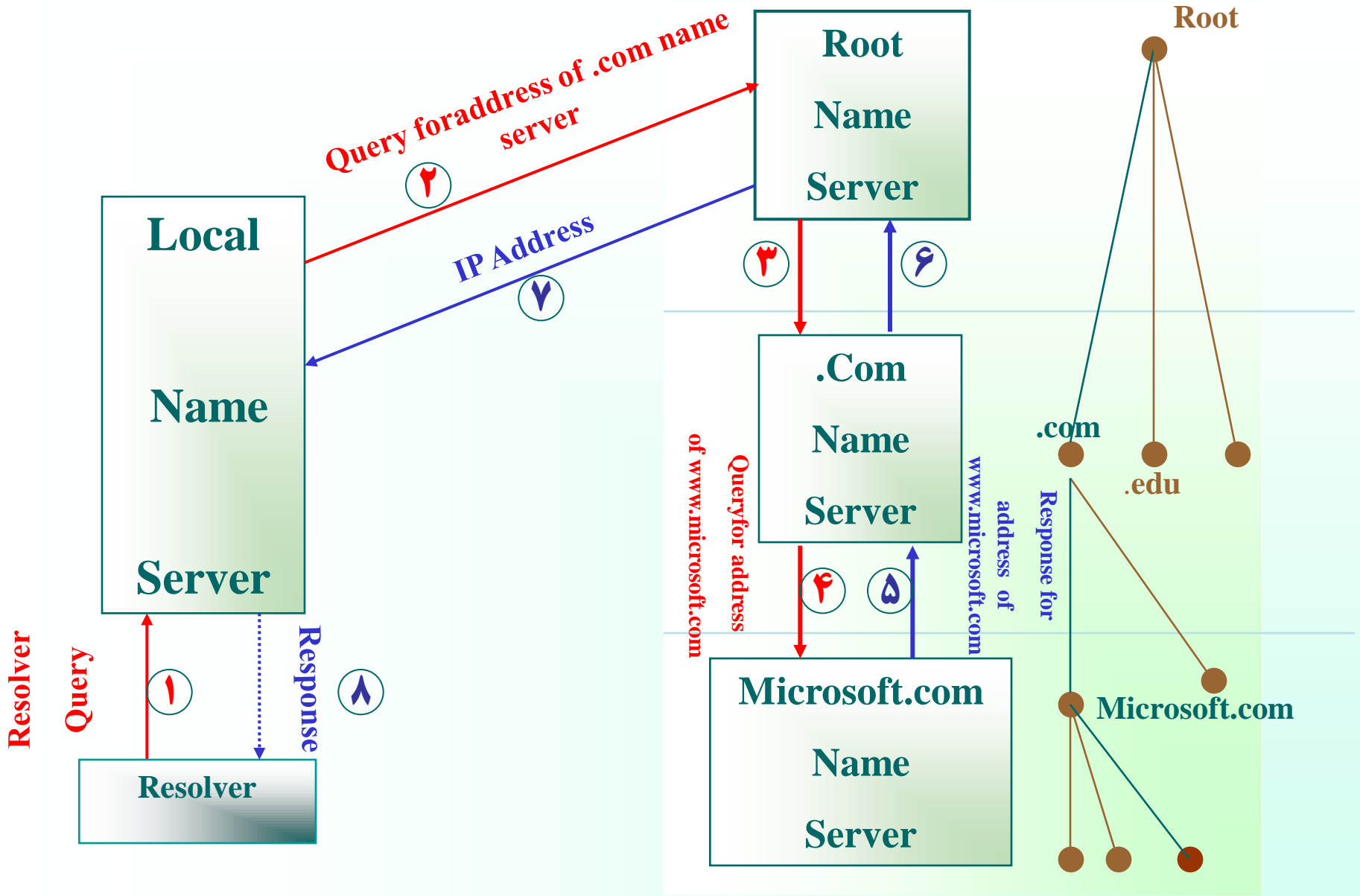
ترجمه نام www.microsoft.com به روش پرس و جوی تکراری



پرس و جوی بازگشتی

- ارسال تقاضای تبدیل نام به روش **UDP** به سرویس دهنده محلی از طرف تابع سیستمی تحلیل نام
- برگرداندن مقدار معادل **IP** در صورت موجود بودن در بانک اطلاعاتی مربوط به سرویس دهنده محلی
- در صورت نبود معادل **IP** نام حوزه در بانک اطلاعاتی سرویس دهنده محلی، ارسال تقاضای ترجمه آدرس توسط خود سرویس دهنده به سرویس دهنده سطح بالاتر
- پیگیری ترجمه آدرس به همین ترتیب توسط سرویس دهنده های سطوح مختلف و به دست آوردن آدرس **IP** معادل

در روش پرس و جوی بازگشتی ماشین سرویس دهنده محلی این مراحل متوالی را نمی بیند و هیچ کاری جز ارسال تقاضای ترجمه یک آدرس برعهده ندارد و پس از ارسال تقاضا برای سرویس دهنده سطح بالا منتظر خواهد ماند.



ترجمه نام www.microsoft.com به روش پرس و جوی بازگشتی

پرس و جوی معکوس

- داشتن آدرس IP یک ماشین و نیاز به پیدا کردن نام نمادین معادل با آن توسط سرویس دهنده DNS
- انجام یک جستجوی وقت گیر و کامل جهت پیدا نمودن نام نمادین

روش کار:

- ارسال یک تقاضا توسط سرویس دهنده محلی برای DNS متناظر با شبکه ای که مشخصه آن در آدرس IP موجود است .
- ارسال تقاضای مربوطه توسط DNS مربوط به شبکه به سرویس دهنده های متناظر با هر زیر شبکه
- برگرداندن نام نمادین حوزه معادل با آدرس IP

ساختار بانک اطلاعاتی سرویس دهنده‌های نام

اجزای سرویس دهنده نام

پروسه سرویس دهنده

بانک اطلاعاتی

پروسه سرویسی دهنده

- برنامه اجرایی جهت پردازش تقاضاهای ترجمه نام از ماشینهای دیگر و ارسال پاسخ مناسب برای تقاضا دهنده
- استاندارد بودن قالب هر تقاضا در شبکه اینترنت جهت ارسال تقاضا و دریافت پاسخ توسط هر ماشین فارغ از ساختار و سیستم عامل آن

بانک اطلاعاتی

- ذخیره داده‌های لازم برای تحلیل یک نام نمادین در بانک اطلاعاتی
- یکسان نبودن ساختار بانک اطلاعاتی در سرویس‌دهنده‌های گوناگون
- بانک اطلاعاتی = بانک رکوردهای منبع = فایل **Resource Records = RR**

فایل RR

- نگهداری در حافظه اصلی جهت بالابردن سرعت جستجو
- فایل متنی
- در نظرگرفتن زمان اعتبار برای هر رکورد درون فایل

نمونه‌های ساختار کوردهای فایل RR

Domain Name	Time to live	Class	Type	Value
-------------	--------------	-------	------	-------

Domain Name	Type	Class	Time to Live	Length	Value
-------------	------	-------	--------------	--------	-------

Domain Name

مشخص کننده نام حوزه یا نام مربوط به یک ماشین (نام نمادین)

Time to Live

نشان دهنده مدت اعتبار رکورد (بر حسب ثانیه)
مقدار فیلد معمولاً ۸۶۴۰۰ ثانیه

Class

این فیلد مشخص می کند که ماهیت نام نمادین مربوط به چه شبکه ای است

کلاس **IN** ← رکورد مربوط به یک نام در شبکه اینترنت

کلاس **CHAOS**

کلاس **Hesiod**

Type

مشخص کننده نوع رکورد

Type	Meaning	Value
SOA	Start of Authority	Parameters for this zone
A	IP address of a host	32-Bit integer
MX	Mail exchange	Priority, domain willing to accept email
NS	Name Server	Name of a server for this domain
CNAME	Canonical name	Domain name
PTR	Pointer	Alias for an IP address
HINFO	Host description	CPU and OS in ASCII
TXT	Text	Uninterpreted ASCII text

انواع رکوردهای اصلی در بانک اطلاعاتی DNS

```

;Authoritative data for cs.vu.nl
cs.vu.nl. 86400 IN SOA star boss (952771,7200,2419200,86400)
cs.vu.nl. 86400 IN TXT "Faculteit wiskunde en informatica"
cs.vu.nl. 86400 IN TXT "Virje universiteit Amsteradam"
cs.vu.nl. 86400 IN MX 1 zephyr.cs.vu.nl.
cs.vu.nl. 86400 IN MX 2 top .cs.vu.nl.

```

```

flits.cs vu.nl. 86400 IN HINFO SUN UNIX
flits.cs vu.nl. 86400 IN A 130.37.231.165
flits.cs vu.nl. 86400 IN A 192.31.231.165
flits.cs vu.nl. 86400 IN MX 1 flits.cs.vu.nl
flits.cs vu.nl. 86400 IN MX 2 zephyr .cs.vu.nl
flits.cs vu.nl. 86400 IN MX 3 top.cs.vu.nl
www.cs.vu.nl. 86400 IN CNAME star.cs.vu.nl
ftp.cs.vu.nl. 86400 IN CNAME zephyr.cs. vy.nl

```

```

rowboat          IN  A      130.37.56.201
                  IN  MX     1 rowboat
                  IN  MX     2 zephyr

```

```

little-sister    IN  A      130.37.62.23
                  IN  HINFO  Mac MacOS

```

```

laserjet          IN  A      192.31.231.216
                  IN  HINFO  "HP LaserJet IISi Proprietary"

```

نمونه فایل RR در یک سرویس دهنده نام

قالب پیامهای پرس و جو در سرویس دهنده های نام

- بخش سرآیند پیام
- بخش پرسش
- بخش پاسخ
- بخش اطلاعات ناحیه
- بخش اطلاعات اضافی

Header

Question

Answer

Authority

Additional

فیلدهای بخش سرآیند پیام

ID							
QR	OpCode	AA	TC	RD	RA	Z	RCODE
QDCOUNT							
ANCOUNT							
NSCOUNT							
ARCOUNT							

فیلدهای بخش پرسش پیام

Domain Name (QNAME)
Type of query (QTYPE)
Class of query (QCLASS)

فیلدهای بخش پاسخ ، اطلاعات ناحیه و بخش اطلاعات اضافی

Name (Variable length)
Type (16 bits)
Class (16 bits)
TTL (32 bits)
Data Length (16 bits)
Data (Variable length)

مقدمه‌ای بر مدیریت شبکه

لزوم بکارگیری پروتکل‌های شبکه

نظارت بر وضعیت شبکه و اجزای آن و همچنین توانایی اعمال مدیریت بر روی ماشینهای میزبان و اجزای یک زیرشبکه (شامل مسیریابها، پلها و ...)

توجه

پیاده‌سازی نرم‌افزارهای مدیریت شبکه در لایه کاربرد جهت مستقل نمودن پروتکل‌های مدیریت از سخت‌افزار شبکه

معماری پروتکل‌های مدیریت شبکه

- تعریف استاندارد مبادله اطلاعات لازم برای نظارت و مدیریت بین ماشینها و مدیر شبکه
- تعریف استاندارد نظارت و کنترل و همچنین تعریف اطلاعات مدیریتی

استانداردهای مدیریت شبکه

- CMOT
- RMON
- SNMPv

مدل SNMP

Simple Network Management Protocol

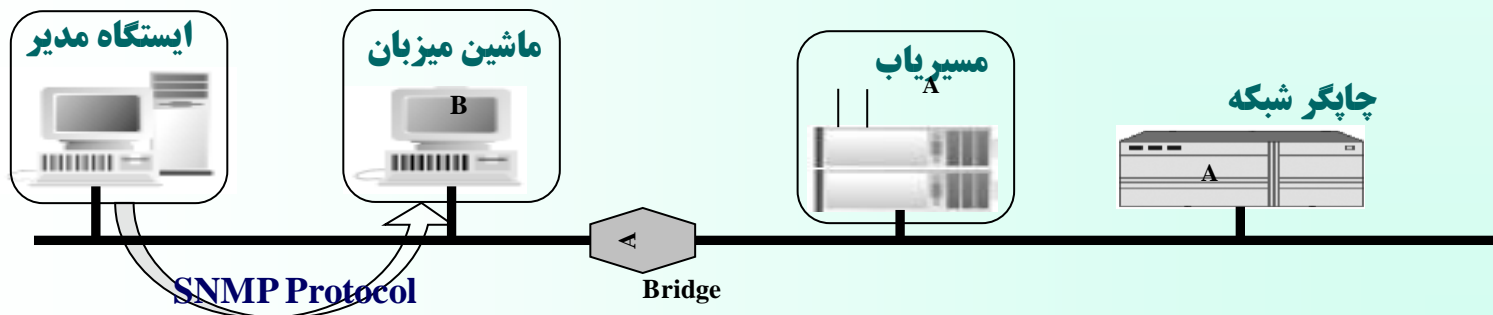
تقسیم عناصر یک شبکه خودمختار به چهار رده:

1. نودهای تحت مدیریت

2. ایستگاههای مدیریت

3. اطلاعات مدیریت

4. قرارداد مدیریت



اجزای مدل مدیریت در SNMP

۱- نودهای تحت مدیریت

- شامل ماشینهای میزبان، مسیریابها، پلها، چاپگرها و هر ماشینی که بتواند اطلاعاتی از وضعیت خود، به ایستگاههای مدیر ارسال نماید و از فرامین آنها تبعیت کند.
- یک نود تحت مدیریت باید قادر به اجرای پروسه کاربردی **SNMP** باشد. در این حالت به آن ایستگاه نمایندگی **SNMP** گفته می شود.
- هر نود تحت مدیریت ممکن است در کنترل چند ایستگاه مدیریت باشد که هر یک از این ایستگاههای مدیر، سطوح دسترسی متفاوتی به آن ایستگاه دارند .

۲- ایستگاههای مدیریت

- مراکز مدیریت شبکه
- کامپیوترهای همه منظوره شامل نرم افزار لازم جهت مدیریت

۳- اطلاعات مدیریت

مشخص کننده وضعیت فعلی ایستگاه (توصیف وضعیت ایستگاه توسط متغیرهای وضعیت در حافظه)

۴- قرارداد مدیریت

روشی استاندارد و مستقل جهت برقراری ارتباط ایستگاه مدیر با نمایندگیها به منظور تقاضای حالت اشیاء (متغیرهای وضعیت) و تغییر آنها در صورت لزوم

لزوم ایجاد استانداردهای مدیریت داده

وجود مجموعه استانداردی از متغیرها برای توصیف وضعیت هر نود تحت مدیریت (از قبیل میزان ترافیک ورودی و خروجی، نرخ خرابی بسته‌های داده، وضعیت اجزای مرتبط و ...).

پایگاه داده اطلاعات مدیریتی Management Information Base

MIB = مجموعه اطلاعات مدیریتی و ساختار پیاده‌سازی آن

استاندارد MIB

- مستقل از پروتکل‌های مدیریت شبکه
- امکان تغییر پروتکل مدیریت ، بدون نیاز به تغییر

MIB

- شامل ۱۰ گروه از اشیاء
- X ستفاده پروتکل‌های مدیریت شبکه از اطلاعات مدیریتی یکسان

Group	# Objects	Description
System	7	Name, location, and description of the equipment
Interfaces	23	Network interfaces and their measured traffic
AT	3	Address translation (deprecated)
IP	42	IP packet statistics
ICMP	26	Statistics about ICMP messages received
TCP	19	TCP algorithms, parameters, and statistics
UDP	6	UDP traffic statistics
EGP	20	Exterior gateway protocol traffic statistics
Transmission	0	Reserved for media-specific MIBs
SNMP	29	SNMP traffic statistics

گروههای اشیاء MIB-II در اینترنت

زبان توصیفی ASN.1

- استاندارد جهت تعریف متغیرهای حالت و اشیاء
- دو مجموعه استاندارد **ASN.1**:
- یک نوع زبان توصیف اشیاء که توسط کاربر قابل استفاده است.
- یک روش کدگذاری برای مبادله اطلاعات بین ایستگاههایی که از پروتکل **SNMP** پشتیبانی می کنند.

پروتکل ساده مدیریت شبکه (SNMP)

به دلیل وجود انواع مختلفی از دستورات در یک پروتکل مدیریت شبکه و در نتیجه پیچیدگی زیاد به جهت اضافه کردن دستورات جدید برای هر نوع عملیاتی



استفاده از روش واکنشی تمامی عملیات و فرمانها و ذخیره متغیرهای حالت در پروتکل
SNMP

Message	Description
Get-request	Requests the value of one or more variables
Get-next-request	Requests the variable following this one
Get-bulk-request	Fetches a large table
Set-request	Updates one or more variables
Inform-request	Manager-to-manager message describing local MIB
SnmpV2-trap	Agent-to-manager trap report

انواع پیغامهای SNMP

1. شماره نسخه پروتکل **SNMP**
2. یک شناسه که گروه ایستگاههای تحت نظارت یک مدیر را مشخص می کند.
3. بخش داده که به چند واحد داده تقسیم می شود.

```
SNMP-Message ::=  
    SEQUENCE {  
        version INTEGER {  
            version-1 (0)  
        },  
        community  
        OCTET STRING,  
        data  
        ANY  
    }
```

قالب پیغام به زبان **ASN**

فصل هفتم: برنامه‌نویسی تحت شبکه اینترنت

Socket Programming

اهداف آموزشی:

- انواع سوکت و مفاهیم آنها
- مفهوم سرویس‌دهنده / مشتری
- توابع مورد استفاده در برنامه سرویس‌دهنده
- توابع مورد استفاده در برنامه مشتری
- معرفی زبان جاوا
- آشنایی با اپلت



روال برقراری ارتباط بین دو برنامه از راه دور:

الف) درخواست برقراری ارتباط با کامپیوتری خاص با IP مشخص و برنامه‌ای روی آن کامپیوتر با آدرس پورت مشخص = درخواست فراخوانی تابع سیستمی **socket()**

ب) مبادله داده‌ها با توابع **send()** و **recv()** در صورت برقراری ارتباط

ج) اتمام ارتباط با فراخوانی تابع **close()**

انواع سوکت و مفاهیم آنها

- سوکتهای نوع استریم = سوکتهای اتصال گرا **Connection Oriented**
- سوکتهای نوع دیتاگرام = سوکتهای بدون اتصال **Connectionless**

سوکتهای نوع استریم مبتنی بر پروتکل **TCP** ← لزوم برقراری یک اتصال قبل از مبادله داده‌ها به روش دست‌تکانی سه‌مرحله‌ای

سوکتهای نوع دیتاگرام مبتنی بر پروتکل **UDP** ← مبادله داده بدون نیاز به برقراری هیچ ارتباط و یا اتصال و عدم تضمینی بررسیدن داده‌ها، صحت داده‌ها و ترتیب داده‌ها

سوکتهای نوع استریم

کاربرد:

پروتکل انتقال فایل FTP

پروتکل انتقال صفحات ابرمتن HTTP

پروتکل انتقال نامه های الکترونیکی SMTP

سوکتهای نوع دیتاگرام

کاربرد:

انتقال صدا و و تصویر یا سیستم

DNS

سوکت socket

- سوکت یک مفهوم انتزاعی از تعریف ارتباط در سطح برنامه‌نویسی
- اعلام آمادگی جهت مبادله داده‌ها توسط برنامه‌نویس به سیستم عامل بدون درگیر شدن با جزئیات پروتکل **TCP** یا **UDP** و تقاضای ایجاد فضا و منابع مورد نیاز جهت برقراری یک ارتباط از سیستم‌عامل

سرویس دهنده / مشتری

تعریف عمومی:

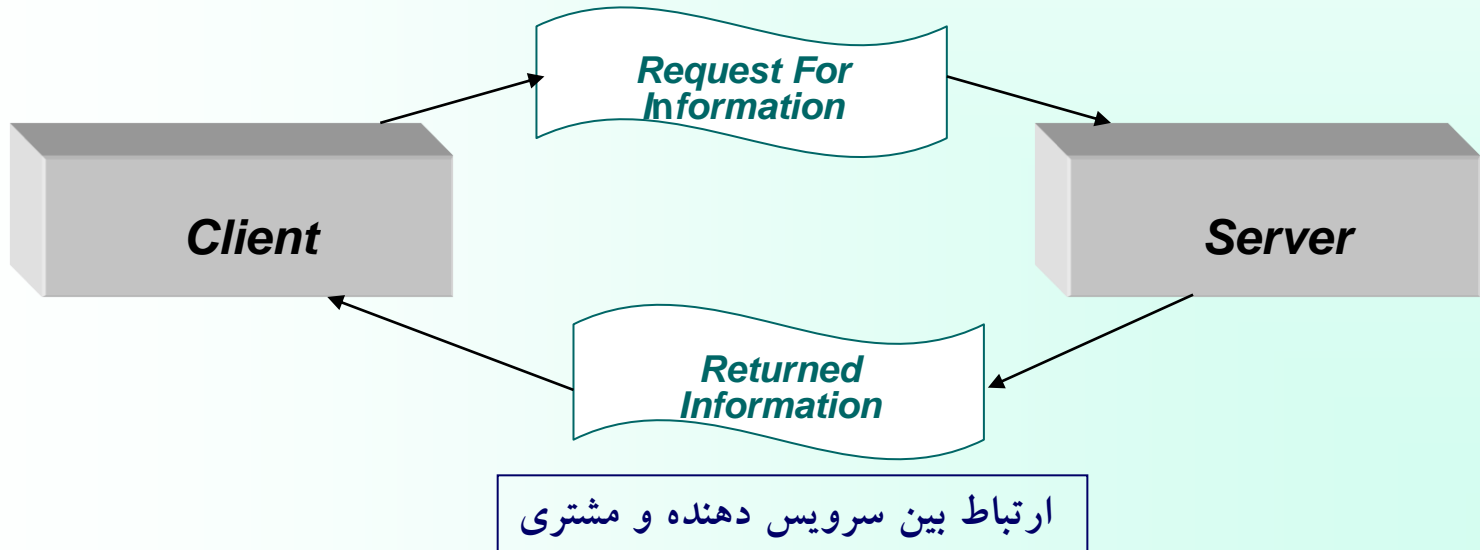
مشتری (client) : پروسه ایست نیازمند اطلاعات
سرویس دهنده (server):
پروسه ای است برای به اشتراک گذاشتن اطلاعات و تحویل اطلاعات به مشتری

برنامه سمت سرویس دهنده Server Side

برنامه‌ای است که روی ماشین سرویس دهنده نصب میشود و منتظر است تا تقاضائی مبنی بر برقراری یک ارتباط دریافت کرده و پس از پردازش آن تقاضا ، پاسخ مناسب را ارسال نماید بنابراین در حالت کلی برنامه سرویس دهنده شروع کننده یک ارتباط نیست.

برنامه های سمت مشتری Client Side

برنامه های سمت مشتری بنا بر نیاز، اقدام به درخواست اطلاعات می نمایند. تعداد مشتریها روی ماشینهای متفاوت یا حتی روی یک ماشین می تواند متعدد باشد و لیکن معمولاً تعداد سرویس دهنده ها یکی است. (مگر در سیستم های توزیع شده)



الف) Socket():

اعلام درخواست ارتباط و تعیین نوع آن (TCP یا UDP) از سیستم عامل با این تابع سیستمی

ب) Bind():

نسبت دادن یک آدرس پورت سوکتی که باز کرده ایم

ج) Listen():

اعلام شروع پذیرش تقاضاهای ارتباط TCP با این تابع به سیستم عامل و تعیین حداکثر تعداد پذیرش ارتباط TCP

د) Accept():

تقاضای معرفی یکی از ارتباطات معلق با استفاده از این تابع از سیستم عامل

ه) Send(),recv():

مبادله داده

و) Close():

قطع ارتباط دو طرفه ارسال و دریافت

ز) Shutdown(): قطع یک طرفه یکی از عملیات ارسال یا دریافت

الگوریتم کار برنامه سمت مشتری

الف) **Socket()**: ایجاد یک سوکت (مشخصه یک ارتباط)

ب) **Connect()**: تقاضای برقراری ارتباط با سرویس دهنده

ج) **Send(),recv()**: ارسال و دریافت داده ها

د) **Close()**: قطع ارتباط بصورت دو طرفه .

Shutdown(): قطع ارتباط بصورت یک طرفه.

توابع مورد استفاده در برنامه سمت سرور (مبتنی بر TCP)

تابع `socket()`

تابع `Bind()`

تابع `Listen()`

تابع `Accept()`

توابع `Send(),recv()`

توابع `Close(),shutdown()`

توابع مورد استفاده در برنامه مشتری (مبتنی بر پروتکل TCP)

تابع `socket()`

تابع `Connect()`

توابع `Send(),recv()`

توابع `Close(),shutdown()`

امکانات زبان جاوا

جاوا زبانی است شیء‌گرا، ساده، ایمن، قابل حمل، توانمند در حمایت از برنامه‌های چند ریسمانی با معماری خنثی

تفاوت‌های زبان جاوا با زبانهای **C, C++**:

- اشاره گرها
- استراکچرها و یونیون‌ها
- توابع
- وراثت چندگانه
- رشته‌ها
- **goto**
- **Operator overloading**
- تبدیل خودکار نوع
- آرگومانهای خط فرمان
- شیء‌گرایی
- مفسر زمان اجرای جاوا

اپلت Applet

- ریزبرنامه یا برنامه کوچکی است که درون یک صفحه وب قرار می‌گیرد و روی یک سرویس‌دهنده اینترنت قابل دسترسی بوده و به عنوان بخشی از یک سند وب بر روی ماشین مشتری اجرا می‌شود.
- برنامه اجرایی است و برای اجرا در محیط مرورگر در نظر گرفته شده تا قابلیت‌هایی که صفحات وب ندارند از طریق آنها فراهم شود.
- اپلت‌ها با برچسب **APPLET** درون صفحه وب تعریف می‌شوند ولی فایلی خارجی به حساب می‌آیند

دو راه اجرای یک اپلت

- اجرانمودن اپلت داخل یک مرورگر سازگار با جاوا مثل **Netscape Navigator**
- استفاده از **Applet Viewer**

محدودیت‌های اپلت

- عدم دسترسی به سیستم فایل جز در موارد محدود
- عدم توانایی در فراخوانی و اجرای برنامه در ماشین اجراکننده آن