
بسم الله الرحمن الرحيم

مبانی برنامه‌سازی سوکت

در شبکه های کامپیو تری ، برنامه های متعددی در یک زمان با یکدیگر مرتبط می گردند.

زمانیکه چندین برنامه بر روی یک کامپیوتر فعال می گردند پروتکل TCP/IP می بایست از روشی به منظور تمایز یک برنامه از برنامه دیگر استفاده نماید بدین منظور از سوکت برای مشخص نمودن یک برنامه خاص استفاده می گردد.

سوکت Socket چیست

با یک بیان ساده می توان گفت که سوکت به ترکیب یک آدرس ماشین (IP) و یک شماره درگاه (Port) گفته می شود.

در برقراری ارتباط بین کامپیوتر ها در یک شبکه دو چیز بسیار مهم است:

۱) آدرس ماشینی که می خواهیم اطلاعاتی از ان بگیریم یا به آن ارسال کنیم.

۲) برنامه ای از آن ماشین که در خواست اطلاعات کرده یا اینکه می خواهیم اطلاعاتی از آن برنامه کسب کنیم.

این دو یعنی آدرس ماشین و شماره برنامه به وسیله سوکت در شبکه مشخص می شوند.

برکلی سوکت

TCP/IP برای اولین بار درسیستم عامل یونیکس معرفی شد و در نگارش های بعدی این سیستم عامل که توسط دانشگاه برکلی توسعه پیدا کرد ، یک رویه برنامه نویسی نیز همراه TCP/IP ارائه شد تا کاربران بتوانند به وسیله آن برنامه های تحت شبکه با استفاده از این پشته پروتکلی ایجاد کنند. این رویه برنامه نویسی به صورت استاندارد برای برنامه نویسی شبکه درآمد و بقیه زبان های توسعه و سیستم عامل نیز از این استاندارد برای پشتیبانی از برنامه نویسی شبکه استفاده کردند.

WinSock

WinSock یا Windows Socket یک رویه (InterFace) برنامه نویسی است که در قالب یک DLL (Dynamic Link Library) در سیستم عامل ویندوز برای برنامه نویسی شبکه و ساخت برنامه هایی که بتوانند با شبکه محاوره داشته باشند معرفی شده است

خدماتی که یک لایه به لایه بالاتر می دهد ممکن است به یکی از گونه های زیر باشد:

- ❖ درخواست سرویس (Request)
- ❖ اقدام لازم برای انجام سرویس (Introduction)
- ❖ ارسال پاسخ سرویس (Response)
- ❖ قبول درخواست (Confirm)

انواع ارتباط لایه های متناظر دو کامپیوتر

ارتباط مابین دو کامپیوتر می تواند به یکی از دو صورت زیر باشد:

- ❖ اتصال گرا (Connection Oriented)
- ❖ غیر اتصال گرا (Connection Less)

در سیستم **اتصال گرا** ابتدا درخواست اتصال ارسال شده و در صورت موافقت طرف مقابل ارتباط برقرار می شود (مثل چیزی که در سیستم تلفن وجود دارد) به این سیستم Data Stream نیز گفته می شود.

اما در سیستم **غیر اتصال گرا** بدون نیاز به موافقت طرف مقابل بسته ها ارسال می شوند (مانند سیستم پست) به این سیستم Data Gram می گویند.

TCP و UDP

در پشته پروتکلی TCP/IP دو نوع ارتباط می توان با کامپیوتر راه‌دور ایجاد کرد:

- ❖ اتصال به کامپیوتر راه دور به وسیله سوکت Data Stream
- ❖ اتصال به کامپیوتر راه دور بدون نیاز به سوکت Data Gram

به بیان غیر رسمی به نوع ارتباط اول (اتصال گرا) ارتباط از نوع TCP گویند و اگر نوع برقرای ارتباط به حالت Data Gram (غیر اتصال گرا) باشد به آن UDP گویند.

کاربرد حالت اتصال گرا در مواقعی است که نیاز به برقراری یک ارتباط امن بین دو ماشین فرستنده و گیرنده داشته باشیم یعنی از صحت دریافت اطلاعات در ماشین گیرنده مطمئن شویم.

اما حالت غیر اتصال گرا (UDP) مواقعی استفاده می شود که خیلی دریافت اطلاعات توسط ماشین گیرنده اهمیتی نداشته باشد.

در زیر انواع ارتباط بین لایه های دو ماشین را می بینید:

- Connection Oriented (Data Stream)
  - Ack (reliable)
    - Message Stream
    - Byte Stream
  - Not Ack (unreliable)
- Connection Less (Data Gram)
  - Ack (reliable)
  - Not Ack (unrilabel)
  - Request & Reply

کیفیت سرویس دهی یک شبکه Quality of service

سرویس که یک شبکه ارائه می دهد باید دارای یک کیفیت خوب و قابل قبول باشد. از لحاظ کیفیت ارائه سرویس شبکه ها دو نوع می باشند:

- کاربر قبل از استفاده از شبکه در مورد کیفیت با شبکه توافق می کند.
- شبکه هیچ تضمینی در مورد کیفیت نکرده ولی سعی می کند حداکثر کیفیت را ارائه دهد.

پارامترهای ارزیابی کیفیت سرویس

۱. مدت زمان ایجاد ارتباط (Connection Establishment).
۲. احتمال قطع ارتباط (Connection Establishment Failure Probability)
۳. پهنای باند عملی قابل استفاده کاربر (Through Put).
۴. زمان انتقال داده (Transit Delay) .
۵. نرخ خطا (Error Ratio) .
۶. امنیت (Security) .
۷. اولویت بندی (Priority) .

مدل هفت لایه OSI

- وقتی مباز به سطوح مختلفی از انتزاع است ، لایه ای باید ایجاد شود.
- هر لایه باید وظیفه مشخص داشته باشد.
- وظیفه هر لایه باید با در نظر گرفتن قراردادهای استاندارد جهاتی انتخاب شود.
- مرزهای لایه باید برای به حداقل رساندن جریان اطلاعات از طریق واسط ها انتخاب شوند.
- تعداد لایه ها باید آنقدر باشد که نیازی به قراردادن وظایف متمایز در یک لایه نباشد.

در ادامه هر لایه از مدل را به نوبت ، با شروع از لایه پایین مورد بحث قرار می دهیم. توجه داشته باشید که خود مدل OSI یک معماری شبکه نیست زیرا خدمات و قراردادهایی را که باید در هر لایه مورد استفاده قرار گیرد را مشخص نمی کند. فقط مشخص می کند که هر لایه چه عملی باید انجام دهد. ISO استاندارد هایی برای تمام لایه ها نیز تولید کرده است، گر چه این ها بخشی از خود مدل مرجع نیستند. هر کدام به عنوان استانداد جهانی منتشر شده اند.

لایه فیزیکی Physical

لایه فیزیکی به انتقال بیت های خام بروی کانال ارتباطی مربوط می شود. اصول طراحی حکم می کند که وقتی بیت ۱ از یک طرف ارسال می شود، در طرف دیگر بیت ۱ دریافت شود، نه بیت صفر. سوال های خاصی که مطرح می شوند عبارتند از : برای نمایش، یک و صفر به چه ولتاژی نیاز است، هر بیت چند نانو ثانیه دوام دارد، آیا انتقال در هر دو جهت به صورت همزمان صورت گیرد، اتصال اولیه چگونه برقرار می شود، وقتی ارتباط دو طرفه قطع شود، اتصال چگونه خاتمه یابد، و واسط شبکه چند پایه دارد و هر پایه به چه منظوری مورد استفاده قرار می گیرد.

در اینجا مدل طراحی با واسط مکانیکی،الکتریکی، و واسط های زمانی و رسانه انتقال فیزیکی که در زیر لایه فیزیکی قرار دارند، سرو کار دارد.

لایه پیوند داده ها Data link

وظیفه اصلی لایه پیوند داده ها این است که با امکانات انتقال اطلاعات خام، خطی را از دید لایه فیزیکی، به خط بدون خطا تبدیل کند. این کار را با شکستن داده های ورودی به قاب های داده (Data Frame) – معمولا به اندازه چند صد بایت یا چند هزار بایت– انتقال ترتیبی قاب ها، و پردازش قاب ها و اعلام وصول قاب هایی که از طرف گیرنده ارسال می شود، انجام می دهد.

مسئله دیگری که در لایه پیوند داده ها وجود دارد، چگونگی حفظ یک فرستنده سریع در دام یگ ماشین گیرنده کند است. برای اینکه انتقال دهنده بداند که گیزنده در آن واحد چه میزان از فضای بافر را در اختیار دارد، باید از راهکار تنظیم ترافیک استفاده شود. غالبا تنظیم ترافیک و پردازش خطا مجتمع می شوند.

شبکه های پخشی مسئله دیگری در لایه پیوند داده ها دارند : چگونگی کنترل دستیابی به کنال مشترک. یعنی اینکه چگونه چندین ماشین که در یک شبکه قرار دارند بتوانند برای ارسال و در یافت از یک کابل مشترک استفاده کنند.

زیر لایه خاصی از لایه پیوند داده ها، به نام زیر لایه کنترل دستیابی به رسانه (MAC) با این مسئله سرو کار دارد.

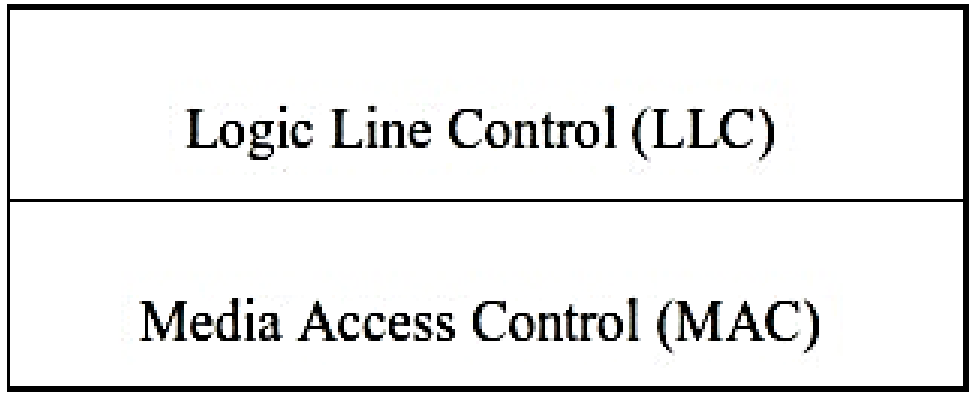
نگاهی دقیق تر به لایه پیوند داده ها و معرفی الگوریتم های مورد استفاده در آن

لایه پیوند داد ها از دو زیر لایه تشکیل شده است (شکل زیر).

این دو زیر لایه از پایین به بالا MAC و LLC نام دارند. در این حالت LLC با لایه Network در ارتباط است و MAC با لایه فیزیکی.

LLC لایه شبکه را قادر می سازد با انواع سرعتها، کابلها و تپولوژی های کار کند.

کار اصلی زیرلایه MAC کنترل دستیابی به لایه فیزیکی است.



نمایش ساختار داخلی لایه پیوند دادها

زیر لایه LLC

وقتی اطلاعات بر روی رسانه ارسال می شود باید با به کار گیری روش هایی مانع از برخورد و ادغام دو بسته اطلاعاتی با هم شویم. یعنی اینکه اگر یک بسته بزرگ به ۱۰ بسته کوچک ۱۰۰ بایتی شکسته شود و این ۱۰ بسته را به ترتیب برای ماشین گیرنده ارسال کنیم کامپیوتر گیرنده با روشی متوجه ابتدا و انتهای بسته اول شود و بسته دوم را نیز تشخیص دهد چون همان طور که می دانید هنگام ارسال اطلاعات روی خط چیزی جز صفر و یک بر روی کابل یا رسانه ارتباطی نیست و باید با روشی ابتدا و انتهای بسته ها برای کامپیوتر راه دور مشخص شود. به کار بستن این مکانیزم ها بر عهده زیر لایه LLC از لایه پیوند داده ها می باشد.

روش های ممانعت از ادغام بسته ها در زیر لایه LLC

- *Character Count.*
- *Starting & Ending Characters , With Character Stuffing.*
- *Starting & Ending Flags , with bit Stuffing.*
- *Physical Coding violation.*

روش اول Character Count

در این روش در اولین بایت یک فریم تعداد بایتهای آن فریم را تععین می کنیم به عنوان مثال اولین بایت فریم زیر مشخص می کند که چهار بایت بعد نیز مربوط به همین فریم است (شکل ۳-۱):

نمایش عملکرد تکنیک Character Count

به این بایت اصطلاحا Character Count می گویند.

ضعف این روش این است که اگر بایت اول آسیب ببیند تا انتهای عمل ارسال تمام اطلاعات به درستی دریافت نمی شوند. و تنها حسن آن هم سادگی روش است.

روش دوم Starting & Ending Characters, with Character Stuffing

در این روش ابتدا و انتهای هر فریم یکسری نشانه اضافه می کنیم یعنی ابتدا و انتها را با چند بایت خاص مش خص می کنیم

DLE برای نشانه گزاری ، STX شروع فریم ، ETX انتهای فریم

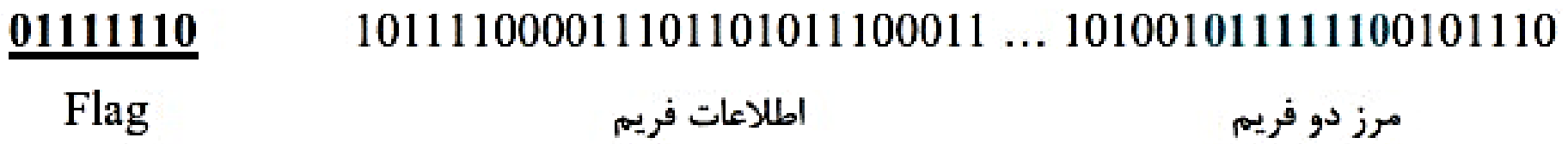
DLE	ETX						DLE	STX

هر گاه که الگوی نشانه گزاری در خود داده های موجود در فریم نیز وجود داشته باشد، باعث ایجاد اختلال در تشخیص ابتدا و انتهای بسته می شود برای حل این مشکل در فرستنده هر گاه الگوی نشانه وجود داشت این الگو را در بدنه دوبار تکرار می شود و در گیرنده اگر دو بار پشت سر هم الگوی نشانه دریافت شود گیرنده متوجه می شود که این جزئی از خود اطلاعات بسته است در نتیجه یکی از آنها را دور ریخته و دیگری را در بسته قرار می دهد.

ایراد این روش در این است که برای مشخص کردن ابتدا و انتهای فریم حداقل به چهار بایت نیاز داریم که این خود باعث به هدر رفتن بخشی از فضای بسته ارسالی می شود.

روش سوم Starting & Ending Flags, with Bit Stuffing

برای مشخص کردن ابتدا و انتهای فریم از یک Flag (پرچم) یک بایتی استفاده می کنند. از لحاظ کارایی با روش Starting & Ending Characters , With Character Stuffing تفاوتی ندارد اما سربار (Over Head) آن کمتر می باشد



ضعف این روش در این است که با تغییر یک بیت ممکن است الگوی flag تغییر کند و کل اطلاعات ارسالی اشتباه دریافت شوند.

روش چهارم Physical coding Violation

در زمان های خاص لبه پایین رونده (قسمتی از شکل موج کلاک سیستم که از بالا به پایین افت ولتاژ می کند) به عنوان یک و لبه بالا رونده به عنوان صفر تلقی خواهد شد. برای مشخص کردن ابتدا و انتهای فریم ها از حالاتی که در کد کردن معنایی ندارد استفاده می شود. یعنی اگر در زمان های مورد نظر تغییر نداشته باشد نه صفر است و نه یک و به این حالت Violation گفته می شود.

اگر به اندازه دو یا سه فریم تغییری در موج نباشد، فرض خواهد شد که پایان فریم است.

این روش بسیار مناسب می باشد و توسط لایه فیزیکی نیز به راحتی قابل اجراست.

معمولا در کاربرد های عملی ترکیبی از روش های گفته شده استفاده می شود (به طور معمول ترکیب روش های ۱ و ۳ و یا ۱ و ۴).

روش های کشف و اصلاح خطا در زیر لایه LLC

در این قسمت الگوریتم ها وشیوه های مختلف تشخیص و اصلاح خطا در یک مجموعه از بیت ها را بررسی میکنیم.

خطا معمولا ناشی از تبدیل یک مقدار در حین انتقال به یک مقدار دیگر است، مثلا تبدیل صفر به یک و یا بلعکس. در شبکه معمولا فقط از مکانیزم های تشخیص خطا استفاده می شود. در صورت مشاهده خطا در خواست ارسال مجدد می کنند و از اصلاح خطا خود داری می شود.

عوامل ایجاد خطا در شبکه

- پارازیت و نویز های خارجی.
- تغییر شکل پالس ارسالی (مثلا به علت تضعیف).

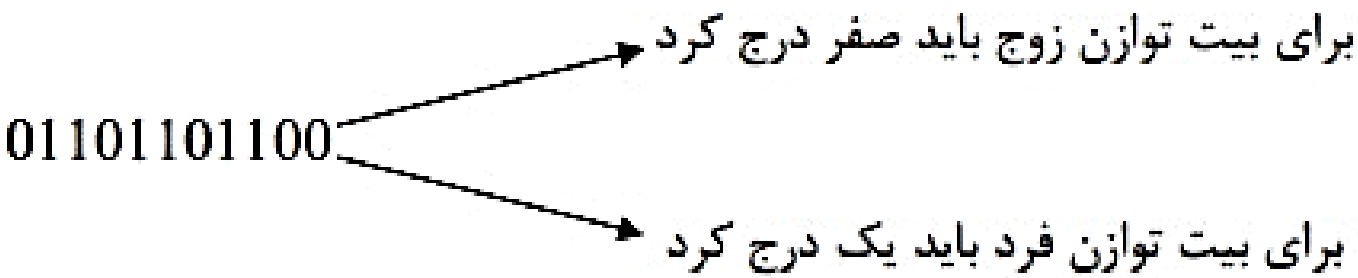
روش های مختلف و گاه متضادی برای تشخیص و اصلاح خطا در شبکه وجود دارد در زیر به مهمترین آنها اشاره شده است :

بیت توازن Parity

ساده ترین روش تشخیص خطا استفاده از یک بیت اضافی به اسم Parity است.

اگر Parity زوج داشته باشیم مقدار این بیت طوری انتخاب خواهد شد که همراه کل مجموعه بیت ها، تعداد بیت های یک زوج باشد. و اگر Parity فرد داشته باشیم طوری بیت های یک را انتخاب می کنیم که تعدا آنها فرد باشد.

مثال:



ضغف این روش در این است که اگر تعداد خطاهای اتفاق افتاد مضربی از دو باشد در (بیت توازن زوج) دیگر نمی توان خطا را تشخیص داد.

روش Hamming Distance برای تشخیص خطا

تعریف فاصله همینگ:

تعداد اختلاف بین بیتهای متناظر در رشته همطول را فاصله همینگ گویند.

مثال:

0110101

1010010

فاصله همینگ دو عدد بالا برابر ۵ می باشد.

نکته:

اگر برای چند رشته بخواهیم فاصله همینگ را بدست آوریم کوچکترین فاصله بین جفت، جفت آنها را به عنوان فاصله همینگ دسته اعداد در نظر می گیریم.

نکته:

اگر فاصله همینگ برابر $d+1$ باشد آنگاه تا d بیت خطا در انها را می توان تشخیص داد.

نکته:

اگر فاصله همینگ برابر $2d+1$ باشد آنگاه تا d خطا را می توان در انها اصلاح کرد.

روش Inter Living

این روش برای ارسال n عدد فریم استفاده می شود. وبه جای ارسال تک تک فریم ها پشت سر هم ابتدا بیت یک کلیه فریم ها را ارسال می کند، و بعد بیت دو و... را ارسال می کند

در این روش از بیت توازن هم برای سطر ها یعنی (فریم ها) و هم برای ستون ها استفاده می شود.

1	2	...	k
1	2	...	k
1	2	...	k
1	2	...	k

حسن این روش در این است که هر گاه در یک ارسال حتی بیش از یک خطا داشته باشیم باز هم برای هر فریم حداکثر یک خطا خواهیم داشت و با استفاده از بیت Parity (توازن) که خطا را چک می کنند، برای هر خانه که خطا داشته باشد هم بیت توازن افقی و هم بیت توازن عمودی خطا را معلوم کرده و در این صورت بیت دارای خطا مشخص و اصلاح می شود.

و ضعف این روش این است که باید تمام فریم ها ارسال شوند تا اطلاعات موجود در گیرنده قابل استفاده باشند.

این روش برای ارسال یک فایل مناسب است اما برای کارهای بلادرنگ (Real Time) مثل ارتباطات صوتی و تصویری ایجاد مشکل می کند. در شبکه تلفن همراه ایران برای کشف خطا از این روش استفاده می شود.

روش CRC

در ای روش فریم داده را به یک Generator تقسیم کرده و باقیمانده تقسیم را به فریم اضافه می کنند و همراه آن به سمت گیرنده ارسال می شود. در گیرنده نیز کل فریم (همراه با باقیمانده) را دوباره بر Generator تقسیم می کنند اگر خطایی روی نداده باشد باقیمانده صفر خواهد بود، در غیر این صورت خطایی در زمان ارسال روی داده است.

این روش محل ایجاد خطا را مشخص نمی کند.

نکته:

در این روش فرستنده و گیرنده باید بر روی Generator توافق داشته باشند.

Generator های معروف:

CRC12 : G = X^12 + X^11 + X^3 + X^2 + X^1 + 1

CRC – CCIT : G = X^16 + X^12 + X^5 + 1 مورد استفاده در شبکه تلفن

CRC16 : G = X^16 + X^15 + X^2 + 1

خطا های زیر را می توان با استفاده از این روش کشف کرد:

- یک یا دو خطا.
- تعداد خطا های فرد.
- خطا های پشت سر هم کمتر از ۱۶.

حسن این روش نسبت به روش های گفته شده دیگر :

- تعداد بیت های اضافه شده تابع طول فریم نیست.
- باقیمانده در انتهای فریم اضافه می شود. این امر باعث پیاده سازی آسان این روش می شود.

نکته:

در کارتهای شبکه امروزی معمولا از روش CRC استفاده می شود، و عمل محاسبه باقیمانده بر عهده کارت شبکه می باشد.

برای اندازه گیری میزان خطا در یک کانال ارتباطی تعداد خطا را در یک میلیون بار ارسال محاسبه کرده و آن را به عنوان شاخص در نظر می گیرند.

معمولا هنگام ارسال بسته از یک hop به hop دیگر باید فرستنده به یک نحوی از رسیدن بسته به مقصد مطلع شود. به همین منظور در لایه پیوند داده ها یکسری پروتکل ها در نظر گرفته شده است. کاربرد این پروتکل ها علاوه بر کشف خطا می تواند قدرت گیرنده را برای دریافت بسته ها نیز در نظر بگیرد.

پروتکل توقف و انتظار Stop & Wait

در این پروتکل فرستنده پس از ارسال یک فریم یک مدت زمانی را منتظر رسیدن پاسخ (Ack) از طرف گیرنده می شود. این کار را با تنظیم یک زمانسنج انجام می دهد. هر گاه زمان تایمر به صفر برسد و پاسخی

برای بسته ارسالی دریافت نشود ، فرض می کند که بسته یا خراب شده یا به مقصد نرسیده است. در این هنگام مبادرت به ارسال مجدد بسته می نماید.

استفاده از این تکنیک در مواردی که فاصله بین فرستنده و گیرنده زیاد است، و همچنین خط دارای درصد خطای بالایی باشد مقرون به صرفه نیست.

پروتکل پنجره های لغزان Sliding Windows

در پروتکل Stop & Wait یک فریم ارسال می شود و سپس فرستنده منتظر رسیدن جواب از گیرنده باقی می ماند، در این حالت مقداری از زمان ماشین فرستنده به بطالت خواهد گذشت. در صورتی که هر ماشین موجود در شبکه بتواند در یک زمان محدود خط را در دست داشته باشد این پروتکل با شکست مواجه خواهد شد.

برای رفع این مشکل پروتکل پنجره های لغزان معرفی شده است، کارکرد این پروتکل به نحوی است که فرستنده هر بار به جای ارسال یک بسته می تواند یک تعداد مشخص از بسته ها (یا اصطلاحا یک پنجره از بسته ها) را ارسال کند و سپس منتظر جواب مربوط به آن تعداد از پنجره ها باشد.

بعد از اینکه فرستنده جواب مربوط به رسیدن صحیح پنجره را از گیرنده دریافت کرد بر روی n تعداد بسته دیگر رفته و آنها را ارسال می کند و دوباره برای دریافت جواب منتظر می ماند.

در این روش گیرنده Ack مربوط به آخرین بسته را ارسال می نماید.

یک نقطه ضعف این روش در این است که اگر یکی از بسته (بجز بسته آخر) در طول عملیات ارسال آسیب ببیند دیگر نمی توان تشخیص داد که کدام بسته بوده است.

برای حل این مشکل از مکانیزم های زیر استفاده می شود:

- ❖ مکانیزم برگشت به n
- ❖ مکانیزم تکرار انتخابی

مکانیزم برگشت به n

در این روش گیرنده اگر تمام فریم های یک پنجره را به طور صحیح دریافت کرد پاسخی برای فرستنده مبنی بر دریافت صحیح اطلاعات ارسال می کند. اما به محض اینکه در یک بسته خطایی را تشخیص داد یا اینکه بسته ای در طول عملیات ارسال گم شود دیگر مابقی بسته ها را نیز دریافت نخواهد کرد و Ack آخرین بسته ای را که صحیح دریافت کرده است را به فرستنده ارسال می کند.

فرستنده نیز با دریافت پاسخ گیرنده متوجه می شود که باید بر روی کدام بسته برود و روند ارسال را از چه بسته ای ادامه دهد.

با کمی دقت بر روی این روش متوجه می شویم که فرستنده باید تا پایان عملیات ارسال تمام بسته ها را در حافظه خود نگه دارد، تا در صورت لزوم مجددا به ارسال آنها مبادرت ورزد.

حسن این روش در پیاده سازی و مدیریت آسان آن می باشد اما همانطور که قبلا نیز گفته شد ممکن است بعد از کشف خطا در یک بسته بقیه بسته های دنباله آن را نیز که خطا ندارد هم دور ریخته شود. که این باعث کندی در عملیات ارسال می شود.

مکانیزم تکرار انتخابی

روش دیگر در برخورد با خطا های روی داده در بسته ها در روش پنجره های لغزان استفاده از مکانیزم تکرار انتخابی می باشد.

در این روش پس از کشف یک خطا در یک بسته دیگر بسته های بعدی دور ریخته نمی شوند بلکه آنها نیز توسط گیرنده دریافت خواهند شد، و برای اینکه بسته دارای خطا دوباره توسط فرستنده ارسال شود در حین دریافت بسته ها با یک Ack به فرستنده اطلاع می دهد که بسته ای که در آن خطا روی داده است را مجددا ارسال نماید.

در این روش از پهنای باند کانال استفاده مطلوب تری صورت می گیرد اما مدیریت بسته ها ، Ack و ترتیب ارسال ، دریافت و چینش بسته ها پیچیده تر می شود.

زیر لایه MAC

وظیفه اصلی لایه MAC کنترل دسترسی ماشین ها به لایه فیزیکی (کانال ارتباطی) است.

هنگام ارسال اطلاعات برروی کابل در صورت نبود مدیریت بر لایه فیزیکی ممکن است مشکلاتی در روند ارسال و دریافت اطلاعات به وجود آید، مثلا ارسال اطلاعات توسط دو یا چند ماشین به صورت همزمان.

قبل از بررسی مکانیزم های موجود در زیر لایه MAC برای مدیریت بر روی رسانه فیزیکی پارامترهای مربوط به یک کانال ارتباطی را معرفی می کنیم:

- Delay (مدت زمانی که یک فرستنده باید صبر کند تا کانال در اختیارش گزارده شود).
- Through Put (هیچ گاه کابل نباید بیکار یا خالی از فریم باشد).

روش ارسال اطلاعات بر روی کابل

همانطور که می دانید در یک شبکه در هر زمان تنها یک ماشین می تواند مبادرت به ارسال اطلاعات نماید. زیرا در صورتی که چند ماشین به صورت همزمان اطلاعات خود را بر روی کانال قرار دهند بسته های اطلاعاتی که به صورت سیگنال های الکتریکی بر روی کانال وجود دارند با یکدیگر برخورد می کنند و از بین می روند.

برای کم کردن Delay و همچنین افزایش Through Put در یک شبکه مکانیزم هایی پیشنهاد شده است که در اینجا نمونه هایی از آنها را بررسی می کنیم.

روش Aloha

در این روش هر کامپیوتر به محض اینکه اطلاعاتی برای ارسال داشته باشد شروع به ارسال آن خواهد نمود.

مزایا:

Delay در این روش صفر است.

معایب:

اگر دو یا چند کامپیوتر همزمان مبادرت به ارسال اطلاعات نمایند بسته ها با یکدیگر تداخل کرده و از بین می روند.

روش الوهای برهه ای Slotted Aloha

برای بهتر کردن روش آلوها زمان انتقال را به قسمت های (برهه های) مجزایی تقسیم می کنند که هر قسمت برای ارسال یک قاب در نظر گرفته می شود.

در این روش اگر ماشینی قصد ارسال اطلاعات را داشته باشد دیگر نمی تواند در هر زمانی اطلاعات را ارسال کند بلکه فقط ارسال اطلاعات در مرز بین دو برهه امکان پذیر است.

در این روش تاخیر ارسال اطلاعات تقریبا برابر نصف زمان یک برهه است.

روش CSMA (Carrier Sense Multiple Access)

کارایی روش های آلوها و آلوهای برهه ای بسیار کم می باشد و این به آن جهت است که هر ایستگاه به دلخواه خود عمل انتقال را انجام می دهد و به عملکرد سایر ایستگاه ها توجهی ندارد. در یک شبکه محلی این امکان وجود دارد که هر ایستگاه تشخیص دهد سایر ایستگاه ها مشغول به چه کاری هستند. بنابراین بر طبق این مکانیزم در فرستنده ای که قصد ارسال اطلاعات را دارد قبل از انجام عمل ارسال ابتدا به خط گوش می دهد اگر خط خالی باشد شروع به ارسال اطلاعات می کند. این نوع ارسال را اصطلاحا CSMA گویند.

برای بالا بردن کارایی در این روش فرستنده می تواند به هر چیزی که ارسال می کند نیز گوش دهد و به محض این که یک تداخل را مشاهده کرد دیگر چیزی ارسال نمی کند. به این مکانیزم CD (Collision Detection) گویند.

پروتکل های مختلفی بر مبنای CSMA/CD طراحی شده اند که در زیر مثال هایی از این نوع پروتکل ها آمده است :

- 1 - Persistent
- non - Persistent
- p - Persistent

پروتکل های بدون اختلال Collision free

با وجود این که در روش CSMA/CD وقتی که یک ماشین کانال ارتباطی را در دست گرفت دیگر اختلالی در ارسال اطلاعات به وجود نمی آید، اما باز هم ممکن است در زمان رقابت برای در دست گرفتن کانال در بین ماشین ها اختلالاتی در روند ارسال به وقوع بپیوندد. و این موضوع برروی کارایی کانال تاثیر منفی می گذارد.خصوصا هنگامی که طول کابل ارتباطی زیاد باشد و همچنین اندازه بسته ها نیز کوچک باشند.

نکته:

روش های متعددی وجود دارد که باعث می شوند که اختلالات در هنگام رقابت برای تصاحب خط از بین برود.

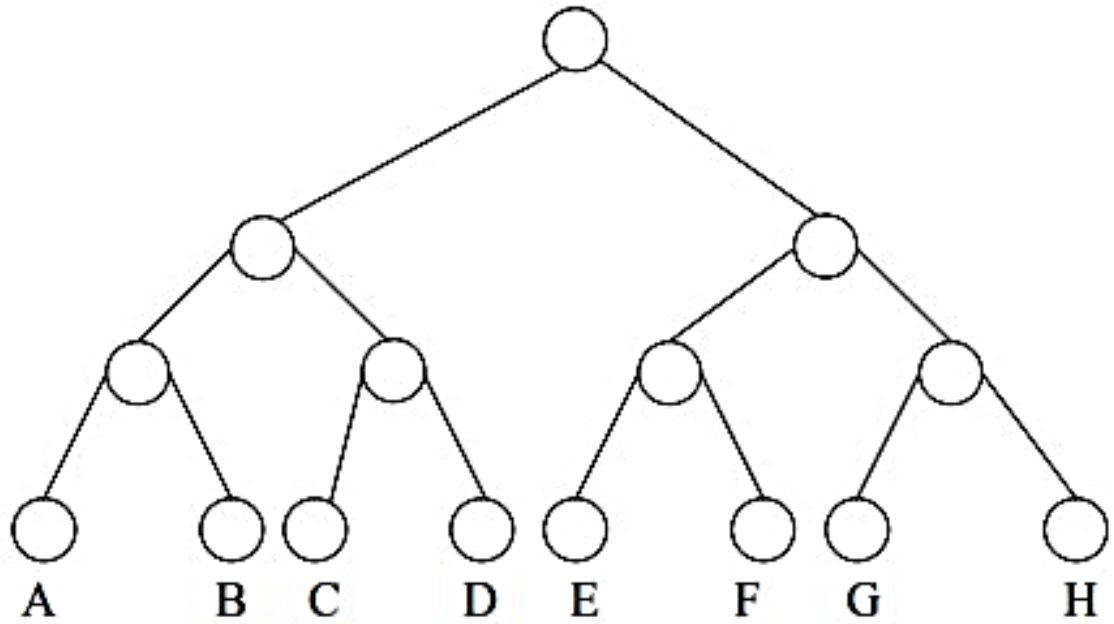
از پروتکل هایی که از مکانیزم Collision Free استفاده می کنند می توان نوع های زیر را نام برد:

- پروتکل نگاشت بیتی
- پروتکل درختی

در اینجا به دلیل اهمییت پروتکل درختی به شرح آن می پردازیم:

در پروتکل درختی از یک درخت دودویی استفاده کرده و ایستگاه های موجود در شبکه را به عنوان برگ های آن در نظر می گیریم، و به صورت زیر عمل می کنیم:

در ابتدا به کلیه ماشین ها (برگ ها) اجازه ارسال اطلاعات داده می شود، اگر فقط یک ماشین داده ای برای ارسال داشته باشد که اختلالی در روند ارسال روی نمی دهد و داده ها به صورت صحیح بر روی کانال ارسال می شوند. اما اگر اختلالی پیش بیاید به زیر درخت چپ فقط اجازه ارسال داده می شود و این روند به صورت بازگشتی ادامه می یابد تا دیگر ماشینی در شبکه نباشد که در عملیات ارسال اختلالی ایجاد کند



ساختار درخت دودویی در شبکه برای پروتکل درختی

لایه شبکه Network

وظیفه لایه شبکه ، کنترل عمل زیر شبکه است. مسئله اصلی در اینجا، تعیین چگونگی هدایت و مسیر یابی بسته ها از منبع به مقصد است.

بسته ها می توانند مبتنی بر جدول های ثابتی باشند که بر مبنای شبکه سیم کشی شده ای است که به ندرت تغییر می کند. آن ها را می توان در آغاز هر عمل ارسال و یا دریافتی مشخص کرد.

در دید دیگر می توان مسیر ها را به صورت پویا مشخص کرد تا در هر بار ارسال بهترین و بهینه ترین مسیر و کم ازدهام ترین مسیر بین فرستنده و گیرنده انتخاب شود.

اگر همزمان بسته های زیادی در زیر شبکه وجود داشته باشند. مسیر عبور را مثل سر بطری تنگ می کنند و عبور مشکل می شود.

کنترل این ازدهام نیز از وظایف لایه شبکه است. بطور کلی، کیفیت، خدمات (تاخیر، زمان انتقال و...) به لایه شبکه مربوط می شود.

وقتی بسته ای برای رسیدن به مقصد مجبور است از شبکه ای به شبکه دیگر برود، مشکلات زیادی ممکن است به وجود آید. ممکن است شیوه آدرس دهی در دو شبکه کتفاوت باشد. ممکن است به علت بزرگی بیش از حد بسته، آن را نپذیرد. پروتکل های ارتباطی ممکن است متفاوت باشند و مشکلات دیگری از این قبیل ممکن است در شبکه ها رخ دهد. از بین بردن این مشکلات برای برقراری اتصال بین دو شبکه ناهمگون، از وظایف لایه شبکه است.

در شبکه های توزیعی، مسئله مسیریابی، چندان مشکل نیست و لذا حضور لایه شبکه بسیار کمرنگ است یا اصلا وجود ندارد.

الگوریتم های مسیریابی

الگوریتم های مسیر یابی که وظیفه آنها هدایت بسته ها از مبدا به مقصد است، قسمتی از نرم افزار لایه شبکه بوده ، که معین می کند بسته رسیده باید به کدام مسیر ارسال شود.

الگوریتم های مسیر یابی مختلفی وجود دارد که در زیر به برخی از آنها اشاره می کنیم:

الگوریتم مسیر یابی کوتاه ترین مسیر (Shorttest Path) :

این الگوریتم بسیار ساده بوده و از ایده ساختن یک گراف از شبکه تبعیت می کند. در این الگوریتم کوتاه ترین مسیر بین مبدا و مقصد (دو گره گراف) پیدا می شود و ارسال اطلاعات بر روی این مسیر صورت می پذیرد.

نکته:

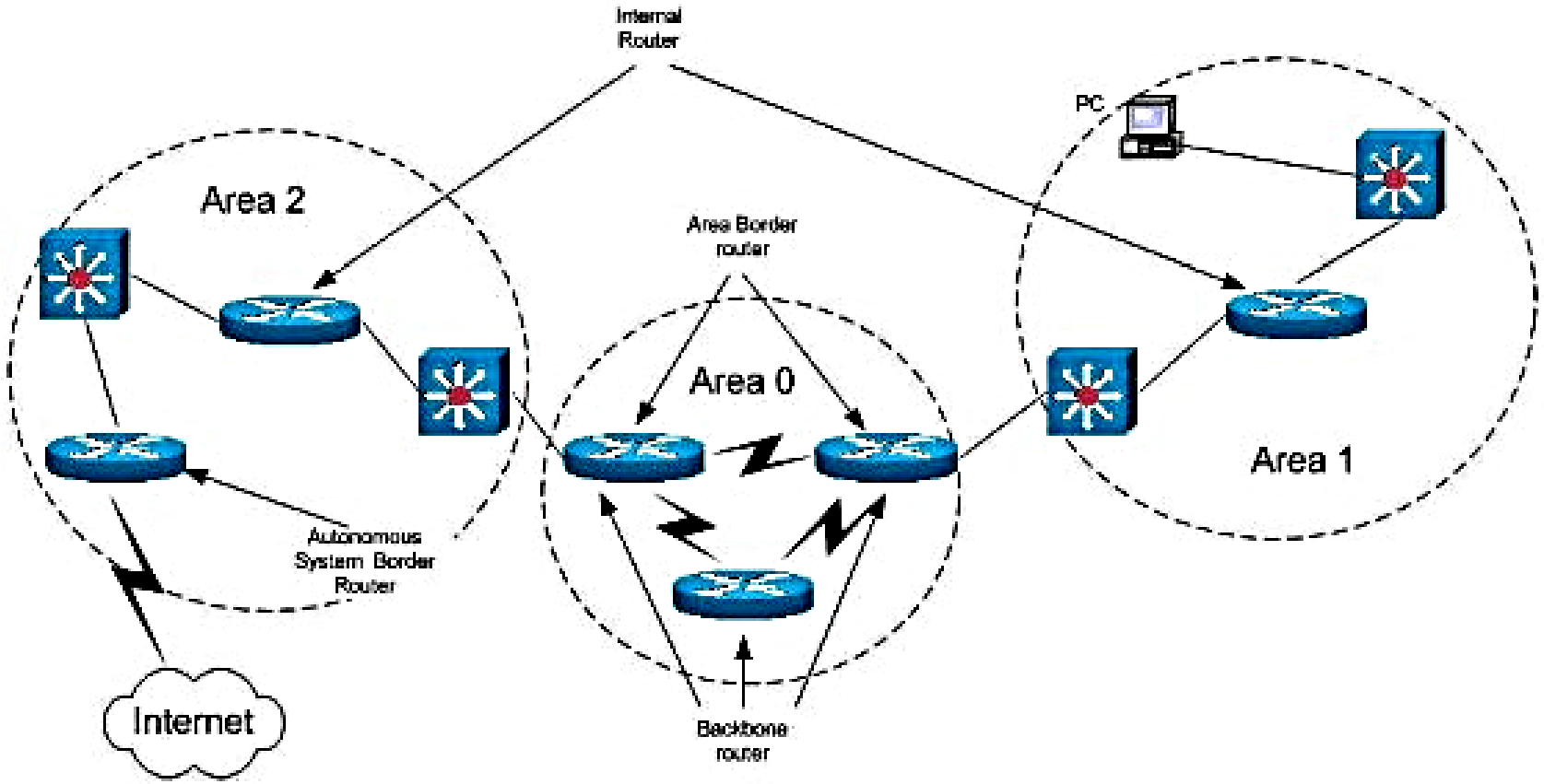
نمونه ای از این الگوریتم، الگوریتم OSPF می باشد که در ماشین های مسیریاب (Router) تجاری استفاده می شود

در یک شبکه مسیریاب ها به دو صورت متوجه می شوند که یکی از مسریابهای همسایه اش دچار خرابی یا اختلال شده است.

روش اول: این روش سخت افزاری است یعنی هرگاه مسیریاب هیچ سیگنال حمل کننده معتبری بر روی خط ارتباطی نبیند فورا متوجه خرابی در خط می شود.این روش در مقایسه با روش های دیگر بسیار سریع است، اما ایراد این روش در این است که ممکن است گاهی سیگنال بر بروی خط موجود باشد اما مسیر یاب مجاور به دلایلی فعالیت نکند و خراب باشد.

روش دوم: در این شیوه بسته ای به عنوان بسته سلام در ابتدای ورود مسیریاب به شبکه برای کلیه مسیریاب های مجاورش ارسال می شود که این بسته حاوی اطلاعات مسیریاب و همچنین هزینه لینک های این مسیریاب

به نقاط دیگر شبکه است. این بسته همچنین در طول مدت فعالیت مسیریاب در فاصله های زمانی مشخصی برای دیگر مسیریاب ها ارسال می شود. عدم دریافت بسته سلام از یک مسیریاب همسایه توسط ماشین مسیریاب شبکه فعلی در فاصله زمانی که برای این منظور تعیین شده است به معنی از کار افتادن مسیریاب همسایه است.



مسیریابی به روش ospf

الگوریتم غرق کن (Flooding) :

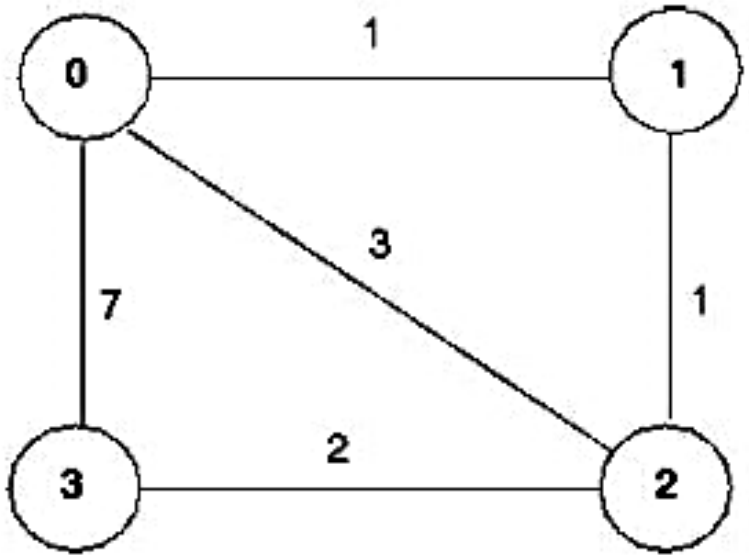
یک الگوریتم مسیریابی می باشد که در آن هر بسته ورودی به تمام خطوط خروجی بجز خطی که از آن آمده است ارسال می شود.

در این الگوریتم نسخه های زیادی از یک بسته در شبکه بوجود خواهد آمد، برای جلوگیری از این مشکل به هر بسته یک شماره نسبت داده می شود، که با عبور از هر مسیریاب یک واحد از آن کم می شود و اگر این شماره به صفر برسد بسته در شبکه از بین می رود. بدین ترتیب از تولید بسته های سرگردان در شبکه جلوگیری می شود.

مسیریابی جریان گرا (Flow-Based Routing) :

در الگوریتم های قبلی فقط به فاصله و تعداد مسیریاب های بین راه اهمییت داده شده بود و به ترافیک موجود بر روی خط های مختلف توجهی نشده بود. اما در این روش مسیریابی، یک گراف برای شبکه تهیه می شود که به هر یال (مسیر) آن عددی اختصاص داده می شود که معرف ترافیک آن مسیر از شبکه می باشد.

بنابراین در این الگوریتم مسیری برای ارسال انتخاب می شود که کمترین ترافیک تا مقصد را داشته باشد (

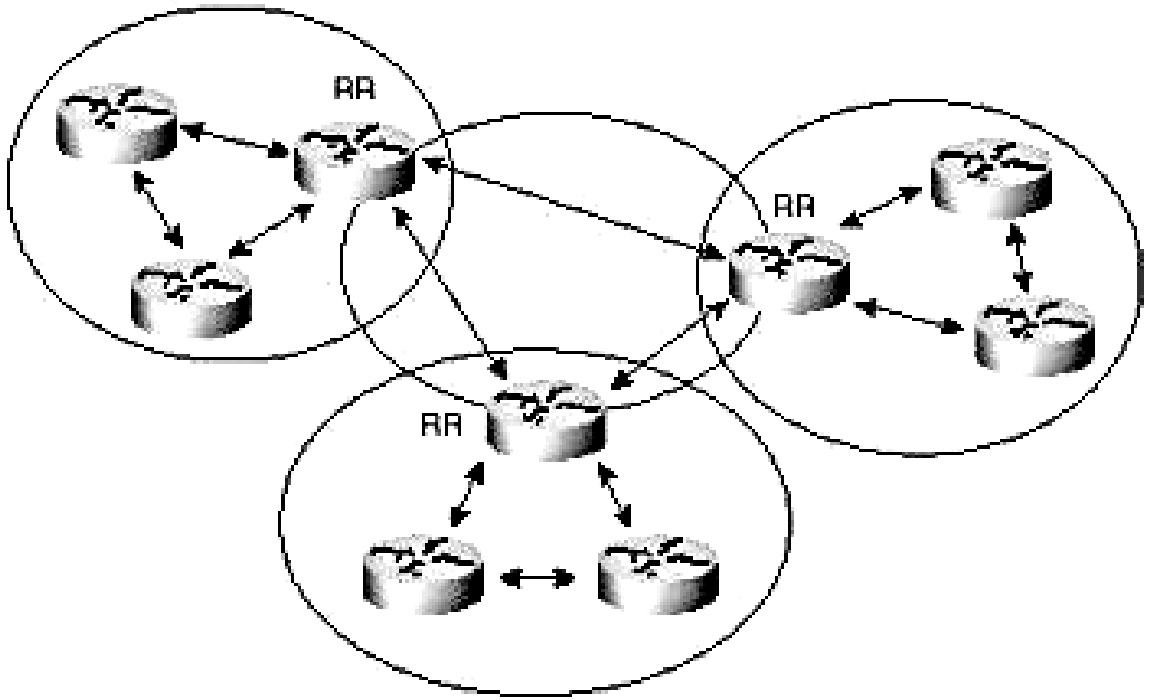


نمایش گراف شبکه که به یالهای آن وزن (هزینه مسیر) اختصاص داده شده است

مسیریابی بردار فاصله :

این الگوریتم مسیریابی در شبکه را به صورت پویا انجام می دهد. در این الگوریتم هر مسیریاب در حافظه خود جدولی یا برداری دارد که در آن بهترین مسیر به هر مقصد را نگهداری می کند و خطی که برای رسیدن به آن مقصد لازم است را مشخص می کند.

برای اینکه این جدول به روز نگاه داشته شود و آخرین تغییرات در آن اعمال شود، مسیریاب های موجود در شبکه در فاصله های زمانی مشخص این جدول را برای یکدیگر ارسال می کنند و همدیگر را از وجود مسیر های شکسته یا مسیر های تازه ایجاد شده مطلع می نمایند

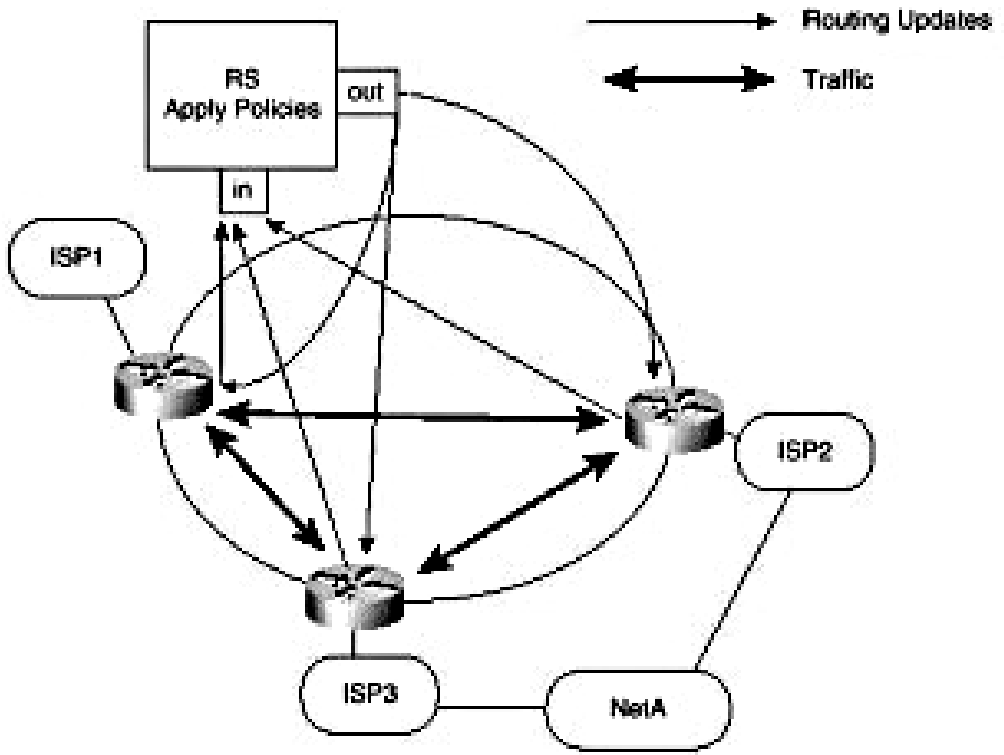


شکل ۱۰-۱:تبادل جدول مسیریابی در بین مسیریاب های موجود در شبکه

مسیریابی حالت پیوند :

الگوریتم بردار فاصله تنها گره های موجود در شبکه و همچنین تاخیر زمانی بین آن ها را مد نظر قرار می دهد. اما ممکن است در یک شبکه گسترده بین گره های مختلف مسیرهایی با پهنای باند متفاوتی وجود داشته باشد. که این موضوع در الگوریتم بردار فاصله نادیده گرفته می شود. برای رفع این مشکل الگوریتم حالت پیوند معرفی شد.

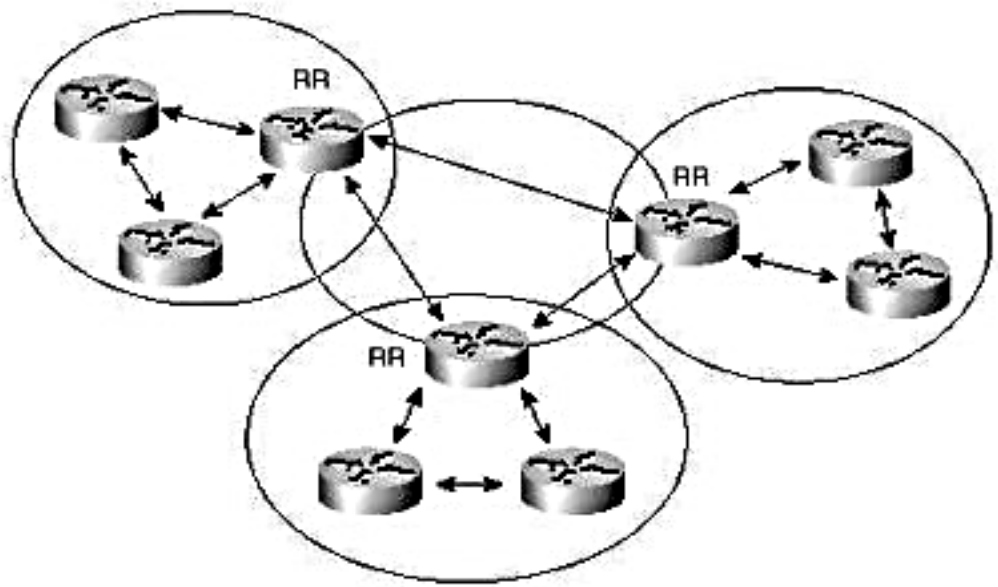
در الگوریتم مسیریابی حالت پیوند برای محاسبه هزینه بین دو نقطه از شبکه پهنای باند موجود بین آن دو نقطه در نظر گرفته می شود . برای این منظور ابتدا مسیریاب های همسایه را تشخیص داده و سپس هزینه (پهنای باند) موجود بین آنها را مشخص می کند و در نهایت این اطلاعات را برای دیگر مسیریاب های مجاورش ارسال می کند



مسیریابی سلسله مراتبی :

با بزرگ شدن روزافزون شبکه ها جداول مسیریابی نیز بزرگتر می شوند. حجم بالای جدول علاوه بر نیاز به حافظه بیشتر باعث صرف زمان بیشتری برای جستجو در جدول مسیریابی می شود.

برای حل این مشکل از تکنیک مسیریابی سلسله مراتبی استفاده می شود. در این تکنیک شبکه به نواحی تقسیم می شود و هر مسیریاب تمام جزئیات مربوط به مسیریابی بهینه را در قسمت مربوط به خود را می داند. هرگاه ماشینی بخواهد اطلاعاتی را به خارج از قسمت خود ارسال کند مسیریاب تقاضای او را به مسیریاب مرزی (مسیریابی که بین دو قسمت مجزا فعالیت می کند) تحویل می دهد و مسیریاب مرزی نیز به نوبه خود بسته را به مسیریاب های سطوح بالاتر که جزئیات بیشتری در مورد شبکه مقصد می دانند تحویل می دهد و این روند ادامه می یابد تا اینکه اطلاعات به شبکه مقصد برسد



نمایش مسیریابی سلسله مراتبی در شبکه های مجزا

مسیریابی سیستم های سیار :

امروزه میلیون ها نفر از کامپیوتر های قابل حمل استفاده کرده ، که علاقه مند به اتصال به شبکه می باشند.

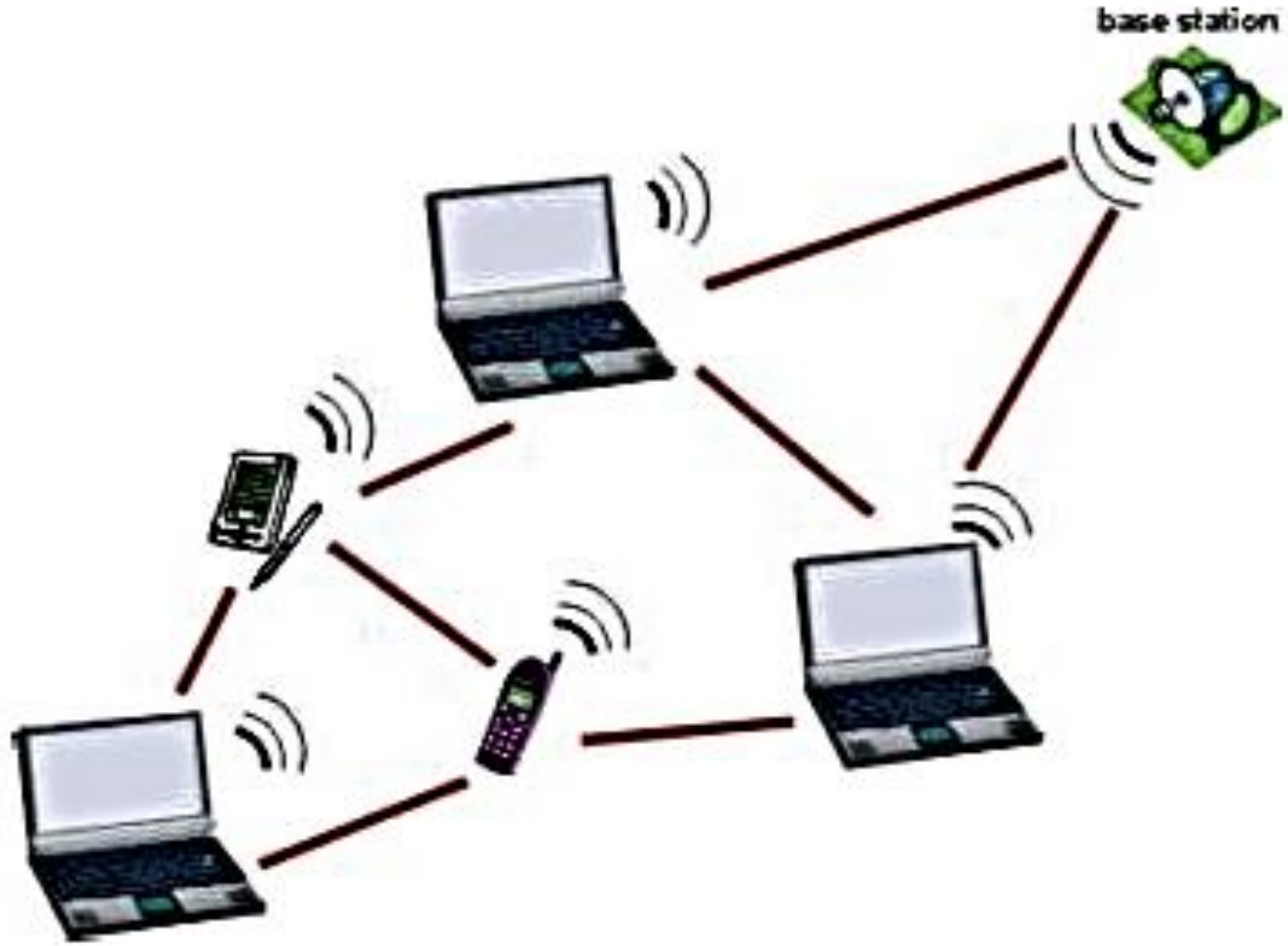
مسیریابی بسته های مربوط به این کامپیوتر ها بسیار پیچیده می باشد. برای این کار باید عمل مسیریابی در هر منطقه توسط مسیریاب آن منطقه انجام گیرد و ضمن حرکت شخص مسیریاب باید عملیات مسیریابی را به نزدیکترین مسیریاب واگذار نماید

از این نوع شبکه ها می توان به شبکه تلفن همراه اشاره کرد که با قرار دادن آنتن هایی در طول مسیر عمل مسیریابی متقاضیان استفاده از سرویس را انجام می دهند.

نکته:

در مسیریابی سیستم های سیار ، ناحیه تحت پوشش به قسمت هایی تقسیم می شود. در هر یک از این قسمت ها مسیریاب یا مسیریاب هایی جهت عملیات مسیریابی در آن ناحیه قرارداده می شوند. وقتی که یک ماشین متحرک درخواست استفاده از خدمات شبکه را می دهد، مسیریاب آن ناحیه ماشین مورد نظر را احراز هویت کرده و در صورت مجاز بودن به استفاده از شبکه، با تعیین سطوح دسترسی به شبکه آن کاربر، اجازه دسترسی به امکانات شبکه را صادر می کند.

در مرحله بعدی سیستم سیار مجاز است که با کد های خاصی که به آن داده می شود، با دیگر قسمت های موجود در شبکه ارتباط برقرار کند. و اگر این ماشین درحین تعامل با شبکه از یک ناحیه به ناحیه دیگری در محدوده سرویس دهی شبکه برود، مسیریاب ناحیه قدیمی اطلاعات آن را به مسیریاب ناحیه جدید که سیستم به آن وارد شده است می دهد. و از آن پس دیگر مسیریاب جدید عملیات مسیر یابی بسته های ارسالی و دریافتی ماشین سیار را به بدست می گیرد. این عملیات بدین صورت ادامه می یابد تا زمانی که کاربر ارتباط خود را با شبکه قطع کند یا اینکه از محدوده سرویس دهی شبکه خارج شود.



مسیریابی ماشین های سیار

آدرس IP :

در شبکه های کامپیوتری مثل اینترنت میلیون ها ماشین فعالیت دارند و مسیریاب ها برای اینکه بداند که بسته ها مربوط به چه ماشینی هستند و باید به کدام ماشین تحویل داده شوند، باید به نحوی ماشین های موجود در شبکه از هم تفکیک شوند. یک راه برای تفکیک ماشین های موجود در شبکه استفاده از یک شناسه یکتاست. این شناسه باید برای هر ماشینی مقداری مستقل داشته باشد تا در شبکه با ماشین های دیگر اختلال نداشته باشد. در شبکه به این شناسه یکتا که معرف ماشینی خاص است آدرس IP گفته می شود.

آدرس IP در لایه سوم مشخص شده است و برای مسیریابی و انتقال بسته ها مورد استفاده روتر های شبکه قرار می گیرد.

نکته :

آدرس IP شامل شماره شبکه و شماره کامپیوتر موجود درآن شبکه است.

آدرس IP بطور استاندارد به صورت چهار عدد در مبنای ۱۰ که با نقطه از هم جدا شده اند نوشته می شود.

مثال:

66.12.201.180

اندازه آدرس IP چهار بایت است و در هر قسمت (هر یک از بایت ها) می توان عدد ۰ تا ۲۵۵ قرار گیرد. دلیل این امر این است که هر بایت از هشت بیت تشکیل شده است و با هشت بیت حداکثر می توان عدد ۲۵۵ را تولید کرد. پس محدوده آدرس های IP می تواند از 0.0.0.0 تا 255.255.255.255 باشد.

نکته:

به دلیل اینکه بخشی از این تعداد آدرس IP برای موارد خاصی در نظر گرفته شده است نمی توان از تمامی طول این رنج برای آدرس دهی شبکه های موجود در جهان استفاده کرد. برای درک بیشتر این موضوع به مثال های زیر توجه کنید:

127.0.0.1

این آدرس LoopBack IP نام دارد و در سیستم های کامپیوتری برای آزمایش و اشکالزدایی از آن استفاده می شود. در صورتی که بسته ای را به این آدرس ارسال کنیم، این بسته تا سطح لایه فیزیکی پایین رفته و دوباره به ماشین برگردانده می شود. بدین ترتیب می توانیم ایرادات احتمالی در سیستم شبکه یک ماشین را کشف کنیم.

255.255.255.255

از این آدرس برای ارسال یک بسته به تمامی ماشین های موجود درشبکه استفاده می شود و به آن Broadcast گویند.

0.0.0.0

این آدرس یک آدرس نامعتبر است و بیشتر در مسیریاب ها برای عملیات مسیریابی و استفاده در پروتکل های مسیریاب از آن استفاده می شود.

192.168.0.0

این آدرس یک Invalid IP است و از آن می توان برای ماشین هایی استفاده کرد که مستقیم به شبکه جهانی اینترنت وصل نیستند (در شبکه های خصوصی مستقل استفاده می شود).

اجزای آدرس IP :

یک آدرس IP از دو بخش تشکیل شده است. بخش اول که مقداری از فضای ۳۲ بیتی آدرس IP را به خود اختصاص می دهد و مشخص کننده شبکه ای است که ماشین به آن تعلق دارد و برای تمام ماشین های موجود در یک شبکه یکسان است. و بخش دوم که آدرس ماشین موجود در شبکه است. این بخش بقیه بیت های موجود در آدرس IP را به خود اختصاص می دهد. بدیهی است که هر چه آدرس شبکه کوچکتر باشد، می توان ماشین های بیشتری را در آن شبکه تعریف کرد و گنجاند و عکس این موضوع نیز درست است یعنی هر چه تعداد بیشتری از بیت های آدرس IP به آدرس دهدی شبکه اختصاص یابد، تعداد کمتری ماشین می توانند در آن شبکه فعالیت کنند.

مثال:

217.219.211.10

217.219.211.50

217.219.211.180

تمام آدرس های فوق مربوط به ماشین های موجود در یک شبکه هستند.

نکته:

به بخش مربوط به مشخصه شبکه در آدرس IP ، NetID گفته می شود.

کلاس های آدرس IP :

در سیستم آدرس IP ما می توانیم ۲ به توان ۳۲ ماشین در جهان را آدرس دهدی کنیم. یعنی حدود چهار میلیارد و سیصد میلیون ماشین. در دنیای شبکه های کامپیوتری برای نظم دادن به شبکه ها و همچنین سرعت بخشیدن به عملیات مسیریابی آدرس های IP را در قالب های خاصی منظم کرده اند که به این قالب ها کلاس آدرس IP گویند.

یک کلاس آدرس IP از بخش های زیر تشکیل شده است:

NetWork Address / SubNet Address / Machine Address

در قسمت اول از سمت چپ آدرس شبکه مشخص می شود در قسمت وسط آدرس زیر شبکه و در نهایت در قسمت سمت راست آدرس ماشین در شبکه بیان می شود.

نکته:

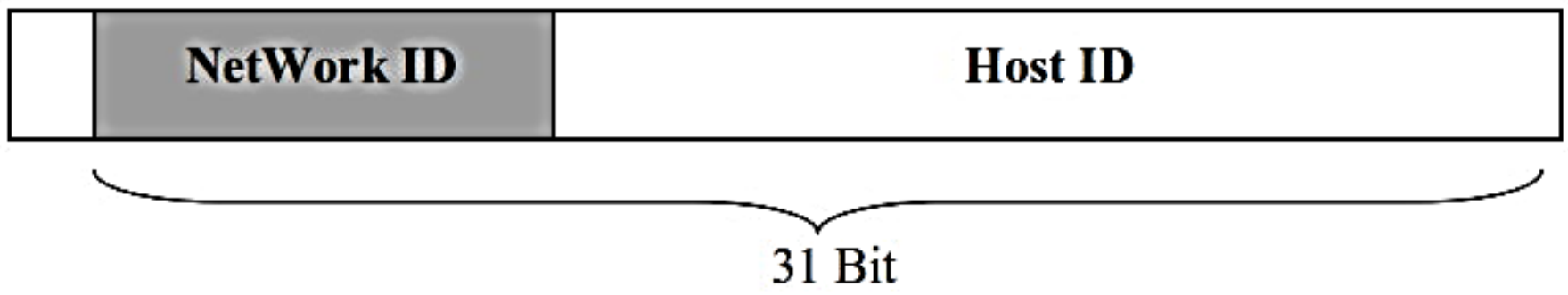
این تقسیم بندی آدرس IP به دستگاه های مسیریاب این امکان را می دهند که به سرعت عملیات مسیریابی را انجام دهند، و بسته ها را به مسیریاب های مناسبی تحویل دهند.

کلاس های پنج گانه:

آدرس IP به پنج کلاس A,B,C,D,E تقسیم می شود.

کلاس نوع A :

در کلاس A اولین بیت سمت چپ با ارزشترین رقم آدرس IP صفر است (شکل ۱۴-۱).



نمایش ساختار آدرس IP در کلاس A

در کلاس A هفت بیت باقیمانده از اولین بایت سمت چپ آدرس شبکه را مشخص می کند و ۲۴ بیت سمت راست آدرس زیر شبکه یا ماشین موجود در این شبکه را نمایش می دهد.

نکته:

به دلیل اینکه برای آدرس دهی شبکه در کلاس A تنها هفت بیت وجود دارد، در کلاس A می توان ۱۲۷ شبکه مجزا تعریف کرد.

نکته:

در کلاس A می توان هفده میلیون ماشین در هر شبکه تعریف کرد. این امر به دلیل این است که برای آدرس دهی یک ماشین از ۲۴ بیت استفاده می شود.

نکته:

آدرس کلاس A به دلیل گستردگی شبکه ای در آن در اختیار شبکه های بزرگ جهانی است.

مثالی از آدرس IP کلاس A :

65.156.35.45

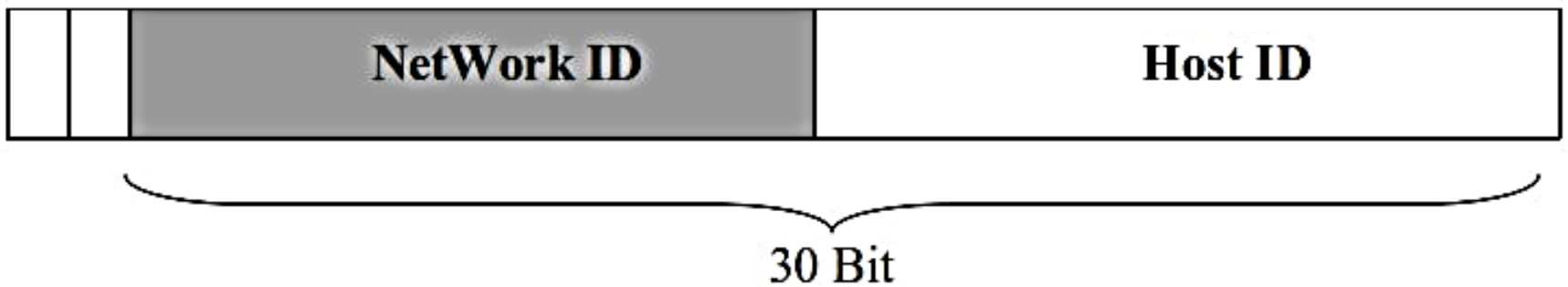
در مثال فوق عدد 65 آدرس شبکه و اعداد 156.35.45 آدرس ماشین است.

نکته:

برای اینکه با دیدن یک آدرس IP تشخیص دهیم که کلاس نوع A است یا خیر، کافیست که به بایت سمت چپ آن نگاه کنیم. اگر عددی بین ۱ تا ۱۲۶ بود آن IP آدرس یک IP در کلاس A است.

کلاس نوع B :

در کلاس B دو بیت سمت چپ پرارزشترین رقم 10 است



نمایش ساختار آدرس IP در کلاس B

در کلاس B برای آدرس دهی شبکه از ۱۴ بیت مورد استفاده قرار می گیرد (از دوبایت سمت چپ ۲ بیت برای مشخص کردن کلاس B و ۱۴ بیت برای مشخص کردن آدرس شبکه).

نکته :

در کلاس B با ۱۴ بیت می توان ۲ به توان ۱۴(شانزده هزار و سیصد و هشتاد و دو) شبکه مختلف تعریف کرد.

نکته :

در کلاس B دو بایت سمت راست برای مشخص کردن آدرس ماشین های موجود در شبکه به کار می رود. با تعداد ۱۶ بیت می توان ۲ به توان ۱۶ بیت ماشین مختلف در شبکه هایی با کلاس IP ، B تعریف کرد.

نکته:

تمام ۱۶ هزار شبکه ای که می توان با کلاس B تعریف کرد، به شبکه های بزرگ در جهان اختصاص داده شده و امروزه دیگر نمی توان شبکه ای با این کلاس ثبت کرد.

مثالی از آدرس دهی در کلاس B :

146.36.45.96

در این مثال عدد 146.36 آدرس شبکه و عدد 45.96 آدرس ماشین است.

نکته:

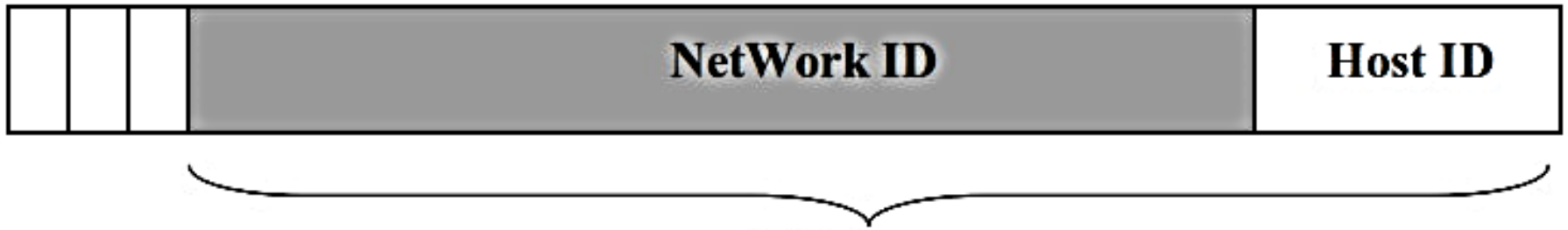
آدرس IP در کلاس B را می توان با مقدار بایت سمت چپ آن تشخیص داد. اگر بایت سمت چپ یک IP عددی بین 128 تا 191 بود، آن آدرس IP یک آدرس در کلاس B است.

آدرس IP با قالب زیر یک آدرس IP در کلاس B است:

128-191.X.X.X

کلاس نوع C :

در کلاس C سه بیت سمت چپ پرارزشترین رقم 110 است



نمایش ساختار آدرس IP در کلاس C

در کلاس C برای بخش آدرس شبکه ۲۱ بیت در نظر گرفته شده است یعنی سه بایت سمت چپ منهای سه بیت که مشخص کننده کلاس C است. و برای قسمت آدرس دهی ماشین ها یک بایت سمت راست رزرو شده ، که می تواند تا ۲۵۴ ماشین در یک شبکه را آدرس دهی کند.

نکته:

در کلاس نوع C به دلیل اینکه ۲۱ بیت برای آدرس دهی شبکه در نظر گرفته شده است، می توان تا حدود دو میلیون شبکه در این کلاس تعریف کرد.

نکته:

در کلاس نوع C تنها در هر شبکه می توان ۲۵۴ ماشین داشت. زیرا برای آدرس دهی ماشین ها در این نوع کلاس فقط می توان از هشت بیت استفاده کرد.

نکته:

کلاس C پراستفاده ترین نوع کلاس در انواع کلاس های آدرس IP است و شرکت های زیادی در دنیا از این نوع کلاس استفاده می کنند.

مثال :

217.219.211.16

در مثال بالا اعداد 217.219.211 آدرس شبکه و عدد 16 آدرس ماشین است.

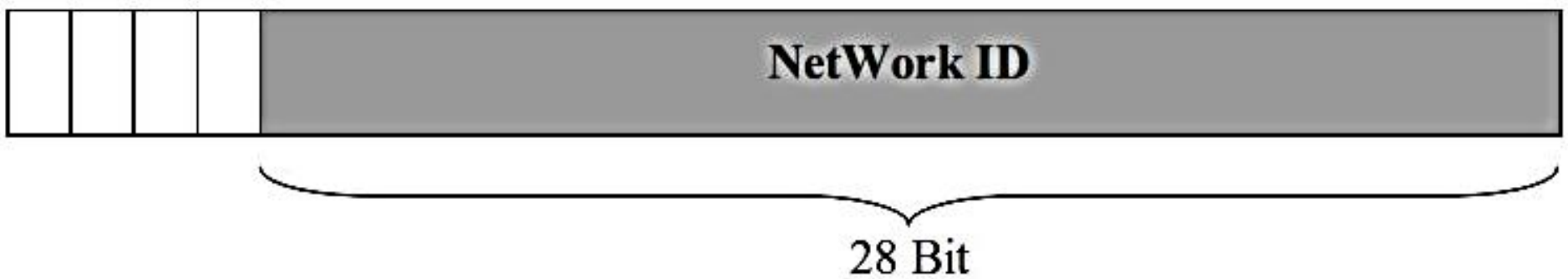
نکته:

در کلاس نوع C بایت سمت چپ آدرس IP می تواند عددی بین 192 تا 223 باشد. یعنی اگر آدرس IP با قالب زیر وجود داشته باشد یک IPاز نوع کلاس C است:

192-223.X.X.X

کلاس نوع D :

در کلاس نوع D چهار بیت پرارزش بایت سمت چپ 1110 است (شکل ۱۷-۱).

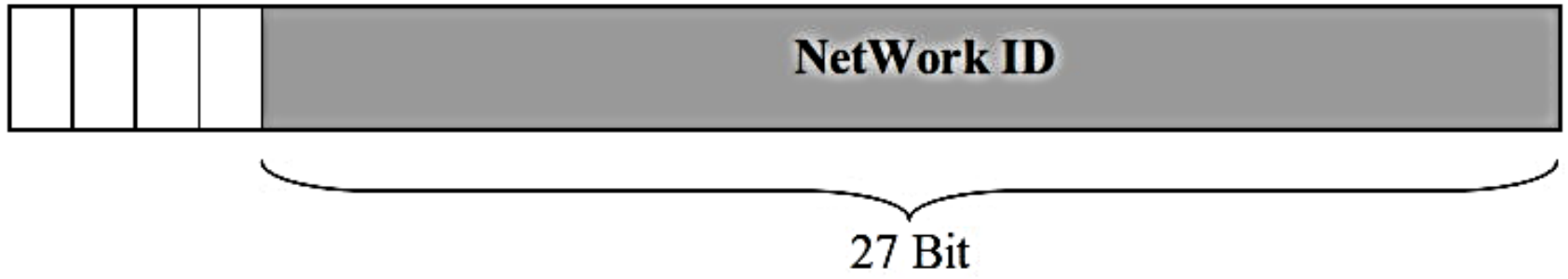


شکل ۱۷-۱ : نمایش ساختار آدرس IP در کلاس D

از این آدرس برای ارسال همزمان به چندین ماشین استفاده می شود.

کلاس نوع E :

در کلاس نوع E پنج بیت پرارزش بایت سمت چپ 11110 است



نمایش ساختار آدرس IP در کلاسE

این کلاس در حال حاضر کاربردی ندارد و برای استفاده در آینده درنظر گرفته شده است.

کنترل ازدحام :

از وظایف دیگر لایه شبکه کنترل ازدحام شبکه می باشد. وقتی که تعداد بسته ها در یک قسمت از شبکه زیاد شود، کارایی آن نقطه از شبکه کاهش می یابد و اصطلاحا گفته می شود ازدحام بوجود آمده است. در این حالت دیگر مسیریاب قادر به مسیریابی بسته ها در شبکه نمی باشد. و بسته ها در مسیریاب از بین خواهند رفت.

لایه شبکه برای جلوگیری و رفع پدیده ازدحام از الگوریتم های مختلفی استفاده می کند که در زیر به تعدادی از آنها اشاره می کنیم:

- ❖ الگوریتم سطل سوراخ دار
- ❖ الگوریتم سطل نشانه
- ❖ الگوریتم تخلیه بار

۴. لایه انتقال (Transport) :

وظیفه اصلی لایه انتقال، پذسرش داده ها از لایه تماس، خرد کردن آن ها به واحد های کوچکتر (در صورت لزوم)، انتقال انها به لایه شبکه و کسب اطمینان ازدریافت صحیح این قطعات در انتهای دیگر است. علاوه بر این، تمام کارها باد به طور بهینه انجام شوند تا لایه های بالاتر را از تغییرات اجباری در تکنولوژی سخت افزار مصون کند.

رایج ترین نوع اتصال در لایه انتقال، تماس نقطه به نقطه (Point To Point) بدون خطاست، که پیام ها یا بایت ها را به ترتیب ارسال، تحویل می دهد.

انواع دیگر خدمات اتصال :

- ❖ انتقال پیام های مستقل بدون تضمین حفظ ترتیب به هنگام تحویل
- ❖ پخش پیام به مقصد های چندگانه

نکته:

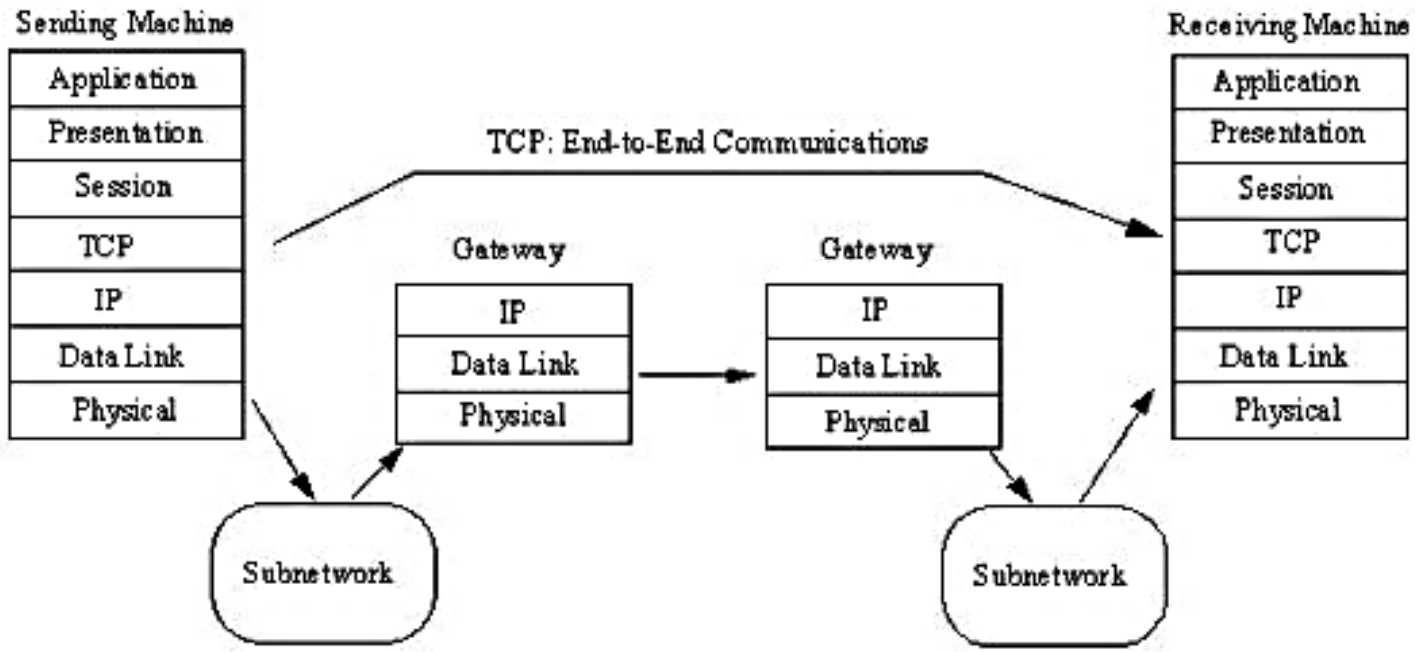
نوع خدمات پس از برقراری اتصال مشخص می شود (دستیابی به خط هایی بدون خطا امکان پذیر نیست. هدف از خط بدون خطا این است که نرخ خطا به حدی باشد که بتوان از آن صرف نظر کرد).

لایه انتقال یک لایه انتها به انتهای واقعی است. به بیان دیگر، برنامه ای در ماشین منبع به کمک هدر پیام و پیام های کنترلی، با برنامه مشابه در ماشین مقصد ارتباط برقرا ر می کند.

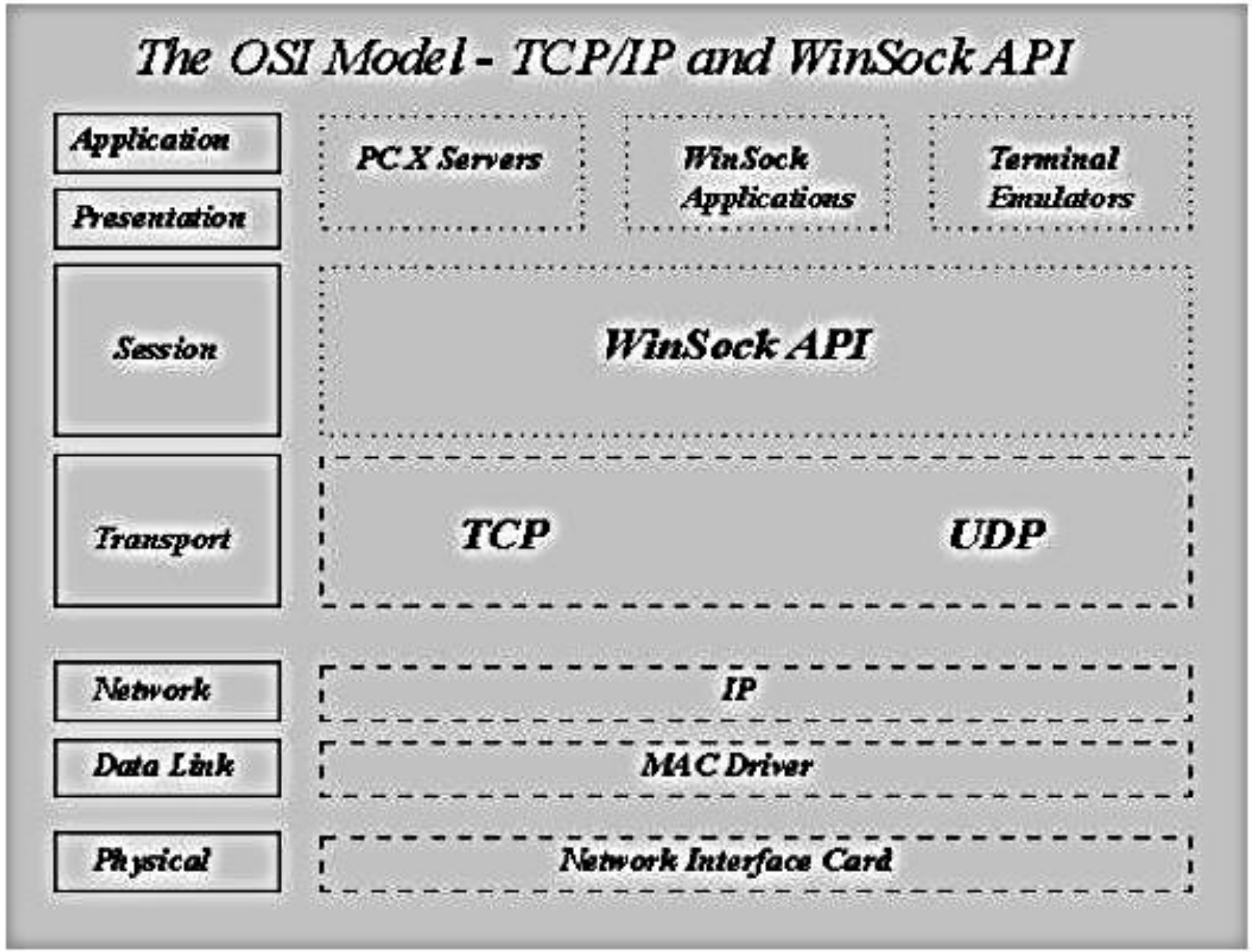
توجه :

لایه های یک تا سه را لایه های گام به گام (Hop to Hop) گویند. یعنی ارتباط در این لایه ها مستقیما با لایه های متناظر در ماشین مقصد برقرار نمی شود بلکه در طول مسیر بین مسیریاب های مختلف تا رسیدن به مقصد به صورت زنجیره ای برقرار است. اما در لایه چهار تا لایه هفت ارتباط انتها به انتها (End to End) وجود دارد یعنی این لایه ها به صورت مستقیم با لایه های متناظر خود در ماشین مقصد ارتباط برقرار می کنند

توجه داشته باشید که مسیریاب ها فقط نیاز به سه لایه اول یعنی لایه های فیزیکی ، پیوند دادها و لایه شبکه دارند. و دیگر به لایه های بالاتر برای انجام وظیفه نیازی ندارند. وقتی بسته ای به یک مسیریاب می رسد ، مسیریاب آن بسته را تحویل مسیریاب بعدی خود می دهد و این روند تا جایی ادامه می یابد که بسته به مقصد برسد وقتی بسته به مقصد رسید، متناظر با نوع بسته بین لایه های بالاتر یک ارتباط انتها به انتها برقرار می شود و تحت این ارتباط دو ماشین به تبادل اطلاعات با یکدیگر می پردازند.



در این لایه مفاهیم سوکت قرار داده شده و برنامه های کاربردی که در لایه هفتم (Application) مدل گنجانده شده اند با استفاده از یک آدرس منحصر به فرد و همچنین یک شماره که مشخص کننده برنامه است با دیگر کامپیوتر ها به تبادل اطلاعات در شبکه می پردازند. در لایه برنامه های کاربردی که لایه شماره هفت از مدل OSI است، ممکن است چندین برنامه مجزا به طور مستقل بر روی شبکه فعالیت داشته باشند یعنی به ارسال و دریافت بسته بر روی شبکه مبادرت کنند. تمام بسته هایی که به یک ماشین وارد می شوند یا بسته هایی که قرار است از ماشین به سوی یک مقصد خاص خارج شوند، تحویل لایه انتقال داده می شوند. لایه انتقال برای اینکه بداند که هر بسته متعلق به کدام برنامه کاربردی در لایه هفتم است، به آن یک شماره مشخصه یکتا می دهد که این شماره می تواند بین ۱ تا عدد ۶۵۵۳۵ باشد. این عدد را اصطلاحا شماره پورت گویند



نمایش مفاهیم مختلف در لایه های ۷ گانه

انواع پورت :

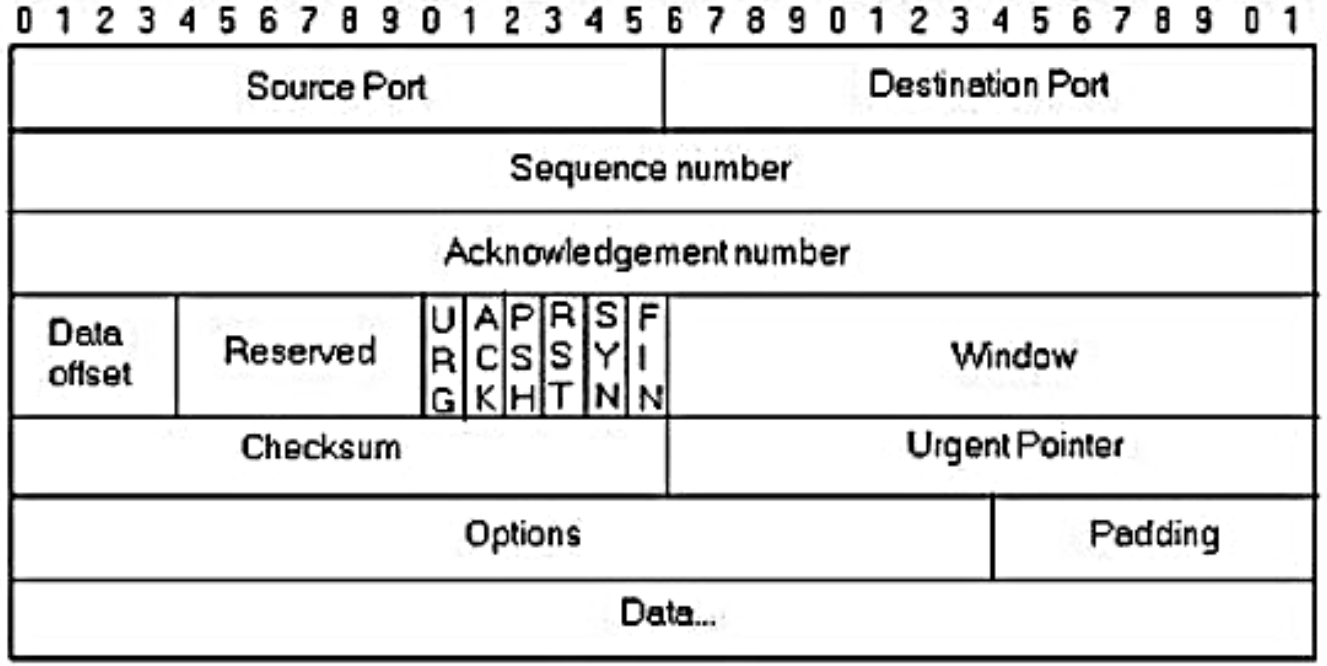
در شبکه بنا به کاربرد های خاص دو نوع پورت وجود دارد:

- ❖ پورت های نوع TCP یا اتصال گرا. این نوع پورت ها برای برقراری ارتباط از پروتکل TCP استفاده می کنند.
- ❖ پورت های نوع UDP یا غیر اتصال گرا. این نوع پورت ها برای برقراری ارتباط از پروتکل UDP استفاده می کنند.

پروتکل TCP :

پروتکل TCP در شبکه برای برقراری یک ارتباط امن بین گیرنده و فرستنده استفاده می شود. این پروتکل به بسته های ارسالی هدری مانند شکل ۲۱-۱ اضافه می کند.

این هدر حاوی فیلد هایی است که فرستنده وگیرنده توسط آن می توانند با یکدیگر ارتباط برقرار کرده و در روند عملیات ارسال و دریافت بسته ها جزئیات کار را به اطلاع هم برسانند.



TCP Header

هدر مربوط به بسته های TCP

در این هدر معانی فیلد ها به شرح زیر است:

- فیلد Source Port شماره پورت برنامه مبدا را مشخص می کند.
- Destination Port شماره پورت برنامه مقصد را نگهداری می کند.

❖ فیلد Sequence Number شماره ای است که به هر بسته برای حفظ ترتیب داده می شود. (از این فیلد برای فرستادن Ack نیز استفاده می شود).

❖ در فیلد Ack Number شماره بسته ای که می خواهیم Ack آنرا ارسال کنیم قرار می گیرد.

❖ TCP Header Length مشخص می کند که طوا هدر TCP چند کلمه ۳۲ بیتی است.

❖ هر گاه بیت URG برابر یک باشد آنگاه مقدار فیلد Urgent Pointer مکان بیت را نسبت به مکان جاری داده های بلادرنگ نشان خواهد داد. (بجای پیام های وقفه مورد استفاده قرار می گیرد)

❖ هر گاه بیت Ack برابر یک باشد نشان می دهد که فیلد Ack Number معتبر بوده و اگر برابر صفر باشد آن فیلد اعتبار نخواهد داشت.

❖ اگر بیت Psh برابر یک باشد یعنی اینکه داده بلافاصله به لایه Application تحویل داده شود (برای مواردی که کاربرد های بلادرنگ مد نظر است) .

❖ اگر بیت Rst برابر یک شود، یعنی در این ارتباط فرستنده مایل است کل عملیات تجدید شود.

❖ اگر بیت Syn صفر باشد ، بدین معناست که اتصال فعال وجود ندارد.

❖ اگر Syn=1 ، Acd=Q یعنی درخواست اتصال.

❖ اگر Syn=1 ، Ack=1 یعنی در خواست اتصال پذیرفته شده است.

❖ برای قط ارتباط بیت FIN برابر یک می شود.

❖ فیلد Check Sum برای کشف خطا در لایه انتقال بکار می رود.

❖ فیلد Option برای این است که امکاناتی که توسط هدر معمولی ارائه نشده ، بتوان به این قسمت اضافه نمود. حداکثر طول این قسمت ۵۳۶ بایت می باشد.

❖ در قسمت Data داده های اصلی بسته جای داده می شوند.

نکته:

ارتباط نوع اتصال گرا بدین معنی است که : مبدا و مقصد برای برقراری ارتباط بین خود، قبل از هر گونه ارسال داده، اقدام به هماهنگی می کنند. و نهایتا هماهنگی به وجود آمده را ختم می کنند و عملیات ارسال و دریافت را خاتمه می دهند.

روش برقراری ارتباط در پروتکل TCP :

برای برقراری ارتباط بین فرستنده و گیرنده در پروتکل TCP از شیوه دست تکانی (Hand Shacking) سه مرحله ای استفاده می شود.

مراحل سه گانه برقراری ارتباط در قرارداد TCP به شرح زیر است:

مرحله اول :

در این مرحله ابتدا از طرف شروع کننده ارتباط یک بسته TCP تهی (بدون اطلاعات) به سمت گیرنده ارسال می شود. در هدر این بسته بیت Syn=1 و فیلد Ack=0 شده است. و همچنین در فیلد Sequence Number شماره ترتیب داده های ارسالی نیز گنجانده شده.

نکته:

شماره فیلد Sequence Number به گیرنده اعلام می کند که شماره بسته های ارسالی از شماره فیلد Sequence Number بعلاوه عدد یک شروع می شود.

نکته:

در شبکه های کامپیوتری برای اینکه مشکلی در روند ارسال اطلاعات بوجود نیاید، شماره Sequence Number از عدد صفر شروع نمی شود. عدد نوشته شده در این فیلد به صورت تصادفی تولید می شود.

مثال :

اگر در فیلد Sequence Number عدد ۶۵ نوشته شود بدین معنی است که داده های ارسالی از طرف فرستنده به گیرنده از شماره ۶۶ شماره گذاری می شوند.

مرحله دوم :

گیرنده با دریافت بسته ای که در هدر آن بیت Syn=1 و بیت Ack=0 شده است، اگر بخواهد ارتباط را برقرار کند یک بسته خالی که در هدر آن بیت های Syn=1 ، Ack=1 و همچنین فیلد Ack Number را برابر مقدار فیلد Sequence Number بعلاوه یک کرده است به سوی تقاضا کنند اتصال، ارسال می کند.

نکته:

اگر گیرنده بسته تقاضای اتصال ، بخواهد تقاضای اتصال را رد کند، بیت Rst را در هدر TCP برابر یک می کند و این بسته را به فرستنده برمی گرداند.

مرحله سوم :

در این مرحله شروع کننده ارتباط با تنظیم کردن فیلد های زیر درهدر یک بسته TCP به گیرنده اطلاع می دهد که آماده ارسال اطلاعات است:

- بیت Syn را یک می کند.
- بیت Ack را برابر یک می کند.
- فیلد Sequence Number را تنظیم می کند.

❖ فیلد Ack Number را برابر مقدار فیلد Sequence Number بسته دریافتی می کند.

پس از اتمام مراحل فوق دو طرف آماده ارسال و دریافت داده می باشند.

خاتمه روند ارسال و دریافت :

در پایان ارتباط هر کدام از طرفین که بخواهند می توانند با تنظیم بیت Fin برابر یک به ارسال اطلاعات به صورت یکطرفه خاتمه دهند ولی در صورتی که طرف مقابل باز هم بسته ای برای ارسال داشته باشد، می تواند این بسته ها را به صورت یک طرفه برای گیرنده ارسال کند.

نکته:

اگر هر کدام از طرفین بر اثر مشکل سخت افزاری یا نرم افزاری ارتباط را بدون هماهنگی با طرف مقابل قطع کند، برای برقراری ارتباط مجدد تا ۱۲۰ ثانیه باید منتظر بماند. این امر به خاطر این است که بسته های ارسالی سرگردان ، ماشینی که بدون هماهنگی قطع شده است نتوانسته آنها را دریافت کند، از زیر شبکه حذف شوند.

در برقراری ارتباط در حالت پروتکل TCP طرفین بعد از دریافت هر بسته ای به صورت صحیح فرستنده را با ارسال یک Ack آگاه می کنند.

پروتکل UDP :

هرگاه داده ها بدون هماهنگی قبلی و بدون مبادله هیچ بسته اطلاع دهنده ای بین دو ماشین گیرنده و فرستنده رد وبدل شود، اصطلاحا به این گونه ارتباط، ارتباط " بدون اتصال یا Connection less " گویند.

نکته:

در پروتکل UDP گیرنده بعد از دریافت هر بسته هیچ اطلاعی به فرستنده مبنی بر درست دریافت کردن بسته یا خرابی آن نمی دهد.

نکته:

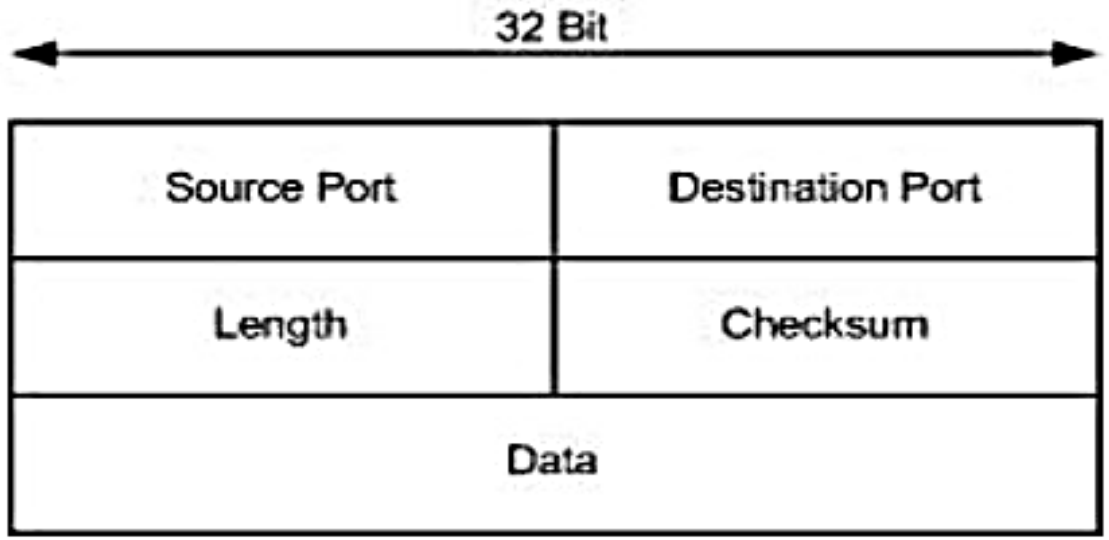
در پروتکل UDP برای اتمام انتقال اطلاعات نیز هیچ پیامی بین فرستنده و گیرنده مبادله نمی شود.

توجه :

پروتکل UDP بدین صورت عمل می کند که، اگر ماشینی بسته ای را بخواهد برای ماشین دیگر ارسال کند، بدون هماهنگی قبیلی شروع به این کار می کند. در طرف مقابل هم ماشین گیرنده اگر بخواهد بسته ها را دریافت می کند و اگر هم نخواهد آنها را دور می ریزد. در طول این روند هم هیچ اطلاعاتی بین فرستنده و گیرنده رد و بدل نخواهد شد.

بسته های پروتکل UDP نیز دارای هدری مخصوص به خود هستند. ولی این هدر در مقایسه به هدر بسته های نوع TCP بسیار ساده تر است، زیرا دیگر نیازی به فیلد ها و بیت های کنترلی مختلف در هدر UDP نیست.

در شکل ۲۲-۱ می توانید هدر مربوط به بسته های UDP را مشاهده کنید:



هدر مربوط به بسته های UDP

در هدر بسته های UDP معنای هر فیلد به شرح زیر است:

- فیلد Source Port شماره پورت برنامه مبدا را مشخص می کند.
- Destination Port شماره پورت برنامه مقصد را نگهداری می کند.
- فیلد Length نیز طول کل بسته را نگهداری می کند.
- فیلد Check Sum برای کشف خطا در لایه انتقال بکار می رود.
- قسمت Data نیز داده های اصلی که باید انتقال داده شوند را نگهداری می کند.