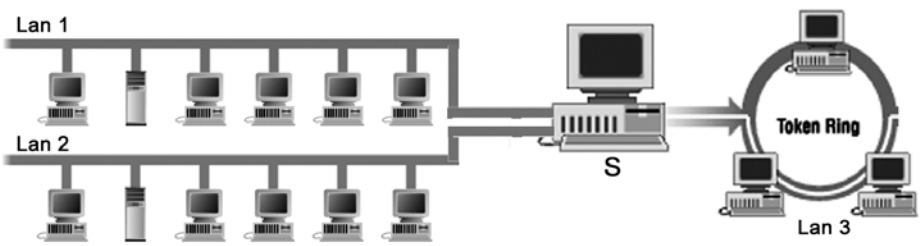
۱) مقدمه

وظیفهٔ لایهٔ اوّل در مدل TCP/IP آنست که یک فریم اطلاعاتی را بین دو کامپیوتر که بر روی یک کانال فیزیکی مشترک واقعند ، منتقل کرده و مسائلی را که در این انتقال ممکن است بروز کند (مثل خطاهای احتمالی کانال) حلّ و فصل نماید. در این نقطه رسالت لایهٔ اوّل پایان می‌یابد.

یک شبکهٔ محلی با توپولوژی Bus را در نظر بگیرید؛ وظیفهٔ لایهٔ اوّل در این شبکه آنست که یک فریم را از ماشینی که به کانال مشترک متصل است به ماشینی دیگر منتقل نماید و از دیدگاه این لایه ماشینی قابل دسترسی خواهد بود که مستقیماً به واسط فیزیکیِ انتقال[۱] متصل شده و آمادهٔ دریافت داده‌ها باشد و در ضمن قواعد شبکهٔ Bus را دقیقاً رعایت نماید. ساختار لایه فیزیکی شدیداً به توپولوژی و سخت‌افزار شبکه وابسته است.

حال ساختار شبکهٔ (۱-۳) را در نظر بگیرید.



شکل (۱-۳) ساختار به هم‌بندی سه شبکهٔ محلی مجزا

در شکل (۱-۳) سه شبکه محلی مستقل وجود دارد که یکی از آنها توپولوژی حلقه و دو شبکهٔ دیگر توپولوژی Bus دارند و هیچگونه ارتباطِ مستقیمِ فیزیکی بین کانالهای انتقال در این سه شبکه وجود ندارد. ماهیت انتقال در شبکهٔ حلقه با نوع BUS متفاوت است و امکان اتصال مستقیم این شبکه‌ها ذاتاً میسّر نیست. تنها نکته قابل توجه در ساختار فوق ایستگاه S است که همزمان به هر سه شبکه متصل است. (ایستگاه S را کامپیوتری با سه عدد کارت شبکه در نظر بگیرید که دوتا از کارتها از نوع Ethernet و یکی از نوع Token Ring است.) S بعنوان عضوی از شبکهٔ حلقه و دقیقاً هماهنگ با پروتکل IEEE 802.5 ، اقدام به ارسال و دریافت اطلاعات روی شبکهٔ محلی LAN3 می‌نماید و همچنین قادر است بعنوان عضوی از دو شبکهٔ BUS

[۱] Medium

اقدام به ارسال و دریافت از شبکه‌های LAN1 و LAN2 کند. سؤالی که مطرح است آنست که : " آیا فارغ از ساختار این سه شبکه، ایستگاه S می‌تواند داده‌هائی را از یک ایستگاه در شبکهٔ LAN1 دریافت کرده و آنرا به ایستگاهی در شبکهٔ LAN3 برساند؟"

جواب مثبت است و رسالت لایهٔ دوم از همین جا آغاز می‌شود یعنی "هدایت بسته‌های اطلاعاتی از شبکه‌ای به شبکهٔ دیگر". در ادبیات شبکه به ایستگاه S ، مسیریاب[۱] و به عمل هدایت بسته‌های اطلاعاتی از مبداء به مقصد مسیریابی[۲] گفته می‌شود.

پس برای ارتباط اطلاعاتی بین دو ایستگاه روی LAN1 و LAN3 (با در نظر گرفتن اختلافِ ساختار فیزیکی دو شبکه) ایستگاه S بوسیلهٔ سخت‌افزار کارت شبکه، داده‌ها را از کانال فیزیکیِ LAN1 دریافت می‌نماید (این داده‌ها در فیلد داده[۳] از فریم لایهٔ فیزیکی شبکه Ethernet قرار گرفته‌اند) و پس از استخراج داده‌ها، مجدّداً آنها را درون فیلد داده از فریم شبکهٔ حلقه قرار داده و روی شبکه تزریق می‌کند.

به قالب فریم در شبکهٔ Ethernet شکل (۲-۳) دقت نمائید.

Preamble	Start of Frame Delimiter	Destination Address	Source Address	Frame Length	Data (Payload)	CRC

شکل (۲-۳) قالب فریم در شبکه Ethernet

فیلدهائی که در فریم چنین شبکه‌ای تعریف شده فقط به این کار می‌آید که فریمی از یک طرف کانال ارسال و طرف دیگر دریافت شود. دو فیلد آدرسی که در این فریم تعریف شده فقط و فقط روی همین شبکه معتبر است و خارج از این شبکه هیچگونه مورد استفاده‌ای ندارند. چرا که وقتی یک واحد اطلاعاتی از یک شبکهٔ محلی به شبکهٔ محلی دیگر منتقل می‌شود ، کلاً قالب فریم و به تبع آن محتوای فیلدهای آدرس باید عوض شود.

به آدرسهائی که در لایه فیزیکی (لایهٔ اوّل) تعریف می‌شود و فقط برای انتقال فریمها روی کانال مورد استفاده هستند "آدرسهای MAC" گفته می‌شود. در حقیقت این آدرسها روشی برای تحریکِ سخت‌افزار کارت شبکه هستند تا اطلاعات را از روی کانال مشترک بردارد. بنابراین چگونگی تعریف این آدرسها و اصول آدرس‌دهی و اندازهٔ این آدرسها (برحسب بایت) شدیداً به ساختار شبکه وابسته است. مثلاً در پروتکل SLIP که ارتباط فقط دوبه دو است اصلاً

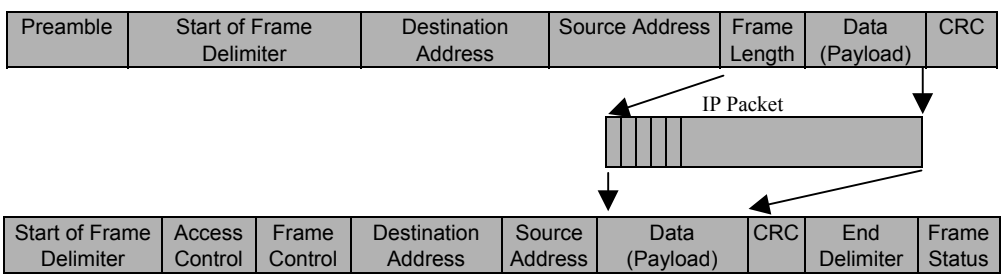
---

۱ Router
۲ Routing
۳ Payload

احتیاجی به فیلد آدرس MAC وجود ندارد در حالی که در پروتکل CSMA/CD (شبکهٔ Ethernet) این آدرسها شش بایتی هستند.

بی‌نظمی در شبکه‌های مختلف و تنوع توپولوژی و پروتکلها و روشهای آدرس‌دهی ، ایجاب می‌کند که برای یکپارچه‌سازی شبکه‌ها و امکان برقراری ارتباط بین آنها در لایهٔ دومِ شبکه تمهیدی اندیشیده شود. اوّلین کار بنیادی ، تعریف آدرسهای جهانی و استاندارد برای تمامی ایستگاههای موجود بر روی شبکه‌های مختلف جهان می‌باشد. در ضمن باید ساختار بسته‌ای که درون فیلدِ داده از فریم هر شبکه قرار می‌گیرد برای تمام شبکه‌ها یکسان و استاندارد باشد بگونه‌ای که هیچ وابستگی به نوع شبکه و سخت‌افزار آن نداشته باشد.

در مدل TCP/IP به واحد اطلاعاتی که باید درون فیلد داده از فریم لایه فیزیکی قرار بگیرد بستهٔ IP[1] گفته می‌شود. تعریف قالب استاندارد یک بستهٔ IP و چگونگی آدرس‌دهی ماشینهای مختلف شبکه و روشهای مسیریابی در لایهٔ دوم از مدل TCP/IP (لایهٔ اینترنت) تعریف شده است. این بسته برای عبور از یک شبکه به شبکهٔ دیگر تغییری نخواهد کرد بلکه از فیلد داده در فریم لایهٔ فیزیکی استخراج شده ، در فریم دیگری قرار می‌گیرد و بدینگونه در شبکه‌ها طی مسیر می‌کند؛ به شکل (۳-۳) دقت کنید. در این شکل فرض شده که یک مسیریاب یک بسته از شبکهٔ Ethernet تحویل گرفته و می‌خواهد آنرا به شبکهٔ حلقه هدایت نماید؛ برای این کار بسته را از فیلد دادهٔ فریمِ شبکهٔ اول استخراج کرده و آنرا درون فریم شبکهٔ دوم قرار داده آنرا ارسال می‌نماید.



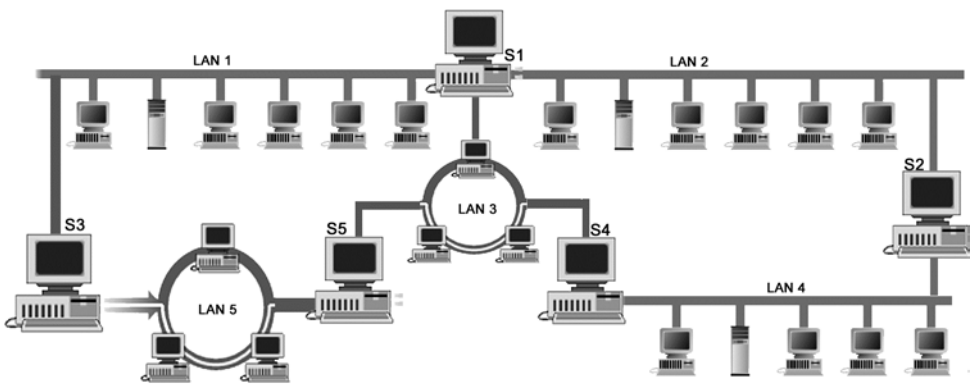
شکل (۳-۳) تغییر قالب فریم و آدرسهای فیزیکی در یک مسیریاب

[1] IP Packet

درون یک بسته تعدادی فیلد به منظور تسهیل در هدایت داده‌ها از یک شبکه به شبکهٔ دیگر در نظر گرفته شده است. دو تا از این فیلدها آدرس مبداء و مقصد هستند که این دو، آدرسهای جهانی محسوب می‌شوند و دو ماشین را فارغ از ساختار شبکه‌ای که به آن متصل هستند بصورت یکتا مشخص می‌کند؛ در شبکهٔ اینترنت به این آدرسها ، آدرسهای IP گفته می‌شود. وقتی یک بستهٔ IP از یک شبکه روی شبکه ای دیگر منتقل می‌شود آدرس‌های MAC (یا کلاً فریم آن) عوض می‌شود ولیکن ساختار بسته ای که درون فیلد داده قرار گرفته و همچنین آدرسهای IP عوض نخواهد شد. در ادامهٔ این فصل ساختار بستهٔ IP را دقیقاً بررسی خواهیم کرد؛ قبل از آن مقدمه‌ای در مورد مسیریاب خواهیم داشت.

۱-۱) مسیریاب

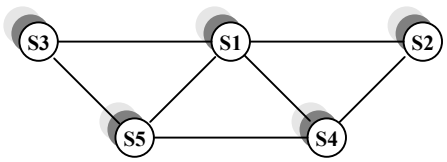
در ساختار شبکهٔ شکل (٤-۳) برای ارتباط دو ماشین روی شبکه‌های متفاوت تنها راه ارتباطی ماشین S است که آنرا مسیریاب نامیدیم. حال ساختار شبکه شکل (٤-۳) را در نظر بگیرید.



شکل (٤-۳) به هم بندی چند شبکهٔ محلی

در این ساختار فرضی S_1 همزمان در شبکه‌های محلی ۱ و ۲ و ۳ حضور دارد؛ S_2 در شبکه‌های ۲ و ٤ ؛ S_3 در شبکه‌های ۵ و ۱ ؛ S_4 در شبکه‌های ٤ و ۳ ؛ S_5 در شبکه‌های ۳ و ۵ حضور دارد. هر یک از ماشینهای S_1 تا S_5 می‌توانند نقش هدایت بسته‌ها را از یک شبکه به شبکهٔ دیگر بر عهده بگیرند. اگر فقط چگونگی ارتباط مسیریابها را با هم در نظر بگیریم و ماشینهای دو شبکهٔ محلی (یعنی ماشینهای میزبان) را که هیچ نقشی در مسیریابی بازی نمی‌کنند حذف نمائیم ، ساختار مسیرها بصورت شکل (۵-۳) در می‌آید. در این شکل کلاً ساختار

شبکه‌های محلی حذف شده و فقط وجود یا عدم وجود ارتباط بین مسیریابها نشان داده شده است. مثلاً اگر چه ارتباط S_1 و S_4 از طریق شبکهٔ محلی LAN3 برقرار می‌شود ولی در شکل (۵-۳) فقط وجود ارتباط (بوسیلهٔ یک خط مستقیم) نشان داده شده است. حال وقتی تعداد شبکه‌های متصل بهم را متعدّد فرض کنید شکل بصورت گرافی پیچیده در خواهد آمد؛ این گراف زیرساخت ارتباطی شبکه‌ها را تشکیل می‌دهد که "زیرشبکه"[۱] نامیده می‌شود.



شکل (۵-۳) زیرساخت ارتباطی مسیریابها

در زیرساخت ارتباطی شبکه‌ها ، مسیرهای متعددی بین دو مسیریاب وجود دارد و بنابراین هر یک از مسیریابها به غیر از وظیفهٔ هدایت بسته‌ها بایستی راهی را برای طی مسیر بسته به سمت مقصد برگزینند که بهینه باشد.

در ساده‌ترین تعریف، مسیریاب ماشینی است که تعدادی ورودی / خروجی داشته و بسته‌های اطلاعاتی را از ورودیها تحویل گرفته و براساس آدرس مقصد، یکی از کانالهای خروجی را برای انتقال بسته انتخاب می‌کند؛ به نحوی که بسته را به مقصد نزدیک نماید.

ماشینی را که هیچ نقشی در هدایت بسته‌های اطلاعاتی روی شبکه ندارد و فقط تولید کننده یا مصرف کنندهٔ بسته‌های اطلاعاتی است ، "ماشین میزبان"[۲] می‌گویند.

از این به بعد هرگاه ساختار زیرشبکهٔ ارتباطی را در قالب یک گراف نشان دادیم، گره‌های این گراف ، مسیریابها را تصویر می‌کند و خطوط بین دو گره در گراف ، کانال ارتباطی بین دو مسیریاب را نشان می‌دهد و بنابراین ماشینهای میزبانِ شبکه را نشان نخواهیم داد.

مجموعهٔ مسیریابها و کانالهای ارتباطی بین آنها ، توپولوژی زیر شبکه را تشکیل می‌دهد و خرابی یکی از مسیریابها یا یکی از کانالهای ارتباطی، توپولوژی زیر شبکه را تغییر داده و در نتیجه ، عملِ مسیریابی در شبکه تحت تأثیر قرار می‌گیرد.

در بیانی ساده ترافیکِ یک کانال ، متوسط تعداد بسته‌های اطلاعاتی است که در واحد زمان روی یکی از کانالهای ورودیِ یک مسیریاب ارسال شده و مسیریاب موظّف به دریافت و

---

[۱] Subnet
[۲] Host Machine

پردازش آن می‌باشد. با توجّه به آنکه تولید بسته‌های اطلاعاتی کاملاً تصادفی و نامعین است بنابراین ترافیک لحظه‌ای هر کانال کاملاً متغیر با زمان خواهد بود.

با این مقدمه ابتدا لایهٔ اینترنت (لایهٔ شبکه) از مدل TCP/IP را معرفی کرده و در فصلی مجزّا به الگوریتمهای مسیریابی در شبکهٔ اینترنت خواهیم پرداخت.

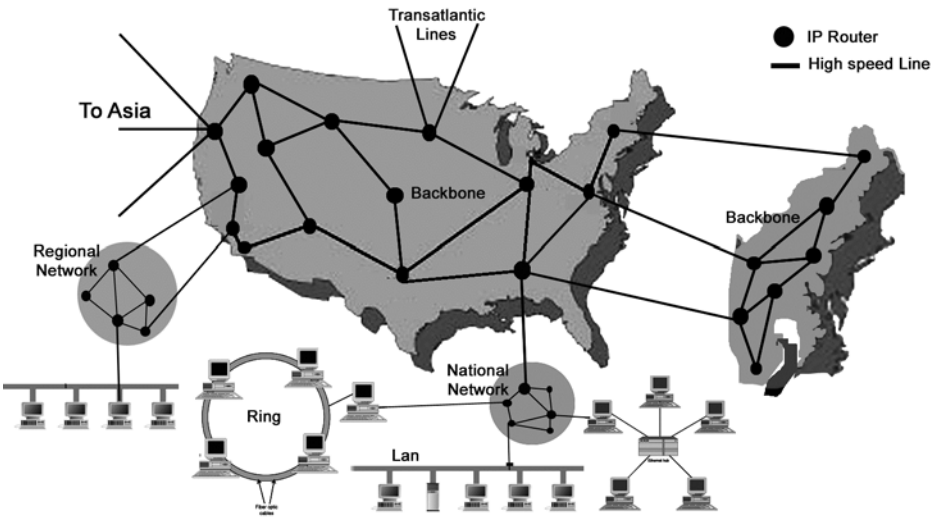
۲) لایهٔ اینترنت

جوهرهٔ اینترنت به گونه ای شکل گرفته است که مجموعه ای از شبکه‌های خودمختار[۱] را به همدیگر وصل می‌نماید . هیچگونه ساختار حقیقی و ثابتی نمی‌توان برای اینترنت متصوّر شد . این نکته را بایستی یادآور شویم که در قسمت "زیرشبکه" از شبکهٔ اینترنت تعدادی از خطوط ارتباطی با پهنای باند (نرخ ارسال) بسیار بالا و مسیریابهای بسیار سریع و هوشمند ، برای پیکرهٔ شبکهٔ جهانیِ اینترنت یک "ستون فقرات"[۲] تشکیل داده است. شبکه‌های منطقه ای و محلّی پیرامون این ستون فقرات شکل گرفته و ترافیک دادهٔ آنها به نحوی از این ستون فقرات خواهد گذشت . ستون فقرات در شبکهٔ اینترنت که با سرمایه گذاری عظیمی در آمریکا، اروپا و قسمتهایی از اقیانوسیه و آسیا ایجاد شده است حجم بسیار وسیعی از بسته‌های اطلاعاتی را در هر ثانیه حمل می‌کنند و اکثر شبکه‌های منطقه‌ای و محلی یا ارائه دهندگان سرویسهای اینترنت[۳] به نحوی با یکی از گره‌های این ستون فقرات در ارتباطند . در شکل (٦-۳) سیمای کلّی و ساده از مفهوم ستون فقرات را می‌بینید.

به گونه ای که در بخش قبلی اشاره شد قراردادی که حمل و تردّد بسته‌های اطلاعاتی و همچنین مسیریابی صحیح آنها را از مبدأ به مقصد ، مدیریّت و سازماندهی می‌نماید پروتکل IP[٤] نام دارد. درحقیقت پروتکل IP که روی تمامی ماشینهای شبکه اینترنت وجود دارد بسته‌های اطلاعاتی را (بسته‌های IP) از مبداء تا مقصد هدایت می‌نماید ، فارغ از آنکه آیا ماشینهای مبدأ و مقصد روی یک شبکه هستند یا چندین شبکهٔ دیگر بین آنها واقع شده است.

---

۱ Autonomous **این اصطلاح در فصل بعدی معرفی شده است.**
۲ Backbone
۳ Internet Service Provider (ISP)
٤ Internet protocol



شکل (٦-٣) سیمای کلّی و تجسّمی ستون فقرات در شبکهٔ اینترنت

ساده ترین تعریف برای پروتکل IP روی شبکهٔ اینترنت بصورت زیر خلاصه می‌شود :

لایهٔ IP یک واحد از داده‌ها را از لایهٔ بالاتر تحویل می‌گیرد؛ به این واحد اطلاعات معمولاً یک "دیتاگرام" گفته می‌شود[1]. امکان دارد طول این دیتاگرام بزرگ باشد ، در چنین موردی لایهٔ IP آنرا به واحدهای کوچکتری که هر کدام "قطعه"[2] نام دارد شکسته و با تشکیل یک بستهٔ IP به ازای هر قطعه ، اطلاعات لازم برای طی مسیر در شبکه را به آنها اضافه میکند و سپس آنها را روی شبکه به جریان می‌اندازد؛ هر مسیریاب با بررسی و پردازش بسته‌ها ، آنها را تا مقصد هدایت می‌کند . هر چند طول یک بسته IP می‌تواند حدّاکثر 64Kbyte باشد و لیکن در عمل عموماً طول بسته‌ها حدود ۱۵۰۰ بایت است . (این قضیه به دلیل آنست که اکثر شبکه‌های محلی دنیا اعمّ از Bus، حلقه ، ستاره ، ... طول فریمی نزدیک به یک تا چند کیلو بایت دارند) پروتکل IP مجبور است هنگام قطعه قطعه کردن یک دیتاگرام ، برای کلّ آن یک شمارهٔ مشخصه و برای هر قطعه یک شمارهٔ ترتیب در نظر بگیرد تا آن دیتاگرام بتواند در مقصد برای تحویل به لایهٔ بالاتر یعنی لایهٔ انتقال بازسازی شود .

---

[1] **اصطلاح دیتاگرام در ادبیّات شبکه‌های کامپیوتری به معانی متفاوت و در موارد متعدّد استفاده شده است. لذا به مورد استفادهٔ آن دقّت داشته باشید.**

[2] Fragment

(مجدداً تأکید می‌کنیم که در این مبحث ، دیتاگرام یک واحد اطلاعات است که به صورت یکجا از لایهٔ IP به لایهٔ انتقال تحویل داده می‌شود یا بالعکس لایه انتقال آنرا جهت ارسال روی شبکه به لایهٔ IP تحویل داده و ممکن است شکسته شود)

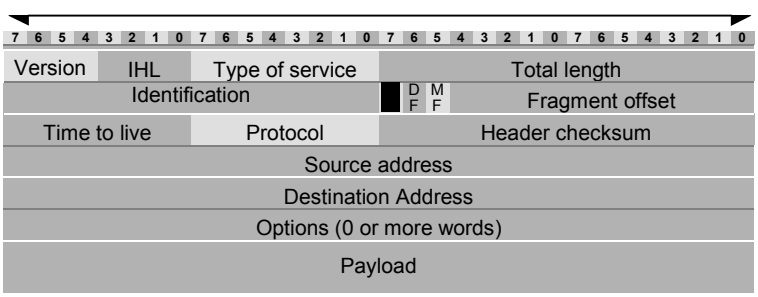
در کنار پروتکل IP چندین پروتکل دیگر مثل ICMP, ARP, RARP, RIP و ... تعریف شده که پروتکل IP را در عملکرد بهتر ، مسیریابی صحیح، مدیریت خطاهای احتمالی یا کشف آدرسهای ناشناخته کمک می‌کنند.

تواناییهایی که پروتکل IP و پروتکلهای جانبی آن عرضه می‌کنند این امکان را فراهم آورده است که تمامی شبکه‌ها و ابزارهای شبکه ای (مثل ماشینهای میزبان ، مسیریابها ، پلها ، و...) فارغ از نوع ماشین و نوع سخت افزار و حتی با وجود تفاوت در سیستم عامل مورد استفادهٔ آنها ، بتوانند بسته‌های IP را با یکدیگر مبادله کنند . پروتکل IP ساختاری استاندارد دارد و به هیچ سخت افزار یا سیستم عامل خاصّ وابسته نیست.

بعنوان اولین گام در شناخت پروتکل IP لازم است قالب یک بستهٔ IP را کالبد شکافی کرده و در گامهای بعدی چگونگی آدرس دهی ماشینها و انواع کلاسهای آدرس در شبکهٔ اینترنت را معرفی نموده و نهایتاً به روشهای مسیریابی و همچنین تعریف پروتکلهای وابسته به IP بپردازیم.

۱-۲) قالب یک بستهٔ IP

شکل (۷-۳) قالب یک بسته IP را به تصویر کشیده است. یک بستهٔ IP از دو قسمت سرآیند و قسمت حمل داده تشکیل شده است. مجموعهٔ اطلاعاتی که در سرآیند بستهٔ IP درج می‌شود توسط مسیریابها مورد استفاده و پردازش قرار می‌گیرد.



شکل (۷-۳) قالب یک بستهٔ IP

➢ فیلد **Version** : اولین فیلد در سرآیند یک بستهٔ IP که چهار بیت است نسخهٔ پروتکلِ IP که این بسته بر اساس آن سازماندهی و ارسال شده است را تعیین می‌کند. در حال حاضر تمامی شبکه‌ها و مسیریابها از نسخهٔ شماره ٤ پروتکل IP پشتیبانی می‌کنند. اگرچه امروزه نسخهٔ شمارهٔ ٦ پروتکل IP به نامهای IPng یا IPv6 معرفی و در حال بررسی و نصب است ولیکن بسیاری از مسیریابها در شبکه‌های دنیا هنوز برای پذیرش این پروتکل آمادگی ندارند و به نظر می‌رسد که تا سال ۲۰۰۵ نگارشِ جدید جهانی نشود. عددی که در حال حاضر در این فیلد قرار می‌گیرد ٤ یا $(0100)_2$ است .

➢ فیلد **IHL**[1] : این فیلد هم چهاربیتی است و طول کل سرآیند بسته را بر مبنای کلمات ۳۲ بیتی مشخص می‌نماید. بعنوان مثال اگر در این فیلد عدد ۱۰ قرار گرفته باشد بدین معناست که کل سرآیند ۳۲۰ بیت معادل چهل بایت خواهد بود . اگر به ساختار یک بستهٔ IP دقّت شود به غیر از فیلد Options که اختیاری است ، وجود تمامی فیلدهای سرآیند الزامی می‌باشد . طول قسمت اجباریِ سرآیند ۲۰ بایت است و بهمین دلیل حداقل عددی که در فیلد IHL قرار می‌گیرد ۵ یا $(0101)_2$ خواهد بود و هر مقدار کمتر از ۵ به عنوان خطا تلقّی شده و منجر به حذف بسته خواهد شد. با توجه به طول ٤ بیتی این فیلد ، بدیهی است که حداکثر مقدار آن ۱۵ یا $(1111)_2$ خواهد بود که در این صورت طول قسمت سرآیند ٦۰ بایت (۱۵×٤) و طول قسمت اختیاری ٤۰ بایت می‌باشد. قسمت اختیاری در سرآیند برای اضافه کردن اطلاعاتی مثل آدرس مسیرهای پیموده شده ، "مهر زمان" و برخی دیگر از گزینه‌هاست که در ادامه توضیح داده خواهد شد.

➢ فیلد **Type of service** : این فیلد هشت بیتی است و توسط آن ماشین میزبان (یعنی ماشین تولید کنندهٔ بستهٔ IP) از مجموعهٔ زیرشبکه (یعنی مجموعهٔ مسیریابهای بین راه) تقاضای سرویس ویژه‌ای برای ارسال یک دیتاگرام می‌نماید . بعنوان مثال ممکن است یک ماشین میزبان بخواهد دیتاگرامِ صدا یا تصویر برای ماشین مقصد ارسال نماید؛ در چنین شرایطی از زیرشبکه تقاضای ارسال سریع و به موقع اطلاعات را دارد نه قابلیت اطمینان صد در صد ، چرا که اگر یک یا چند بیت از داده‌های

[1] IP Header Length

ارسالی در مسیر دچـار خرابـی شـود تـاثیر چندانـی در کیفیـت کـار نخواهـد گذاشـت ولـی اگـر بسـته‌های حـاوی اطلاعـات صـدا یـا تصویـر بـه سـرعت و سـر موقـع تحویـل نشـود اشکال عمـده بوجـود خواهـد آمـد. در چنیـن مواقعـی ماشـین میزبـان از زیرشـبکه تقاضـای سـرویس سـریع (و لاجـرم غـیر قـابل اطمینـان) می‌نمـاید. در برخـی از محیطـهای دیگـر مثـل ارسـال نامـهٔ الکـترونیکی یـا مبادلـهٔ فـایل انتظـار اطمینـان[1] صـد درصـد از زیـر شـبکه وجود دارد و سرعت تاثیر چندانـی بـر کیفیّـت کـار نـدارد .

از طریق این فیلد نوع سرویس درخواستی مشخص می‌شود، این فیلد خودش به چنـد بخـش تقسیم شده است :

P2	P1	P0	D	T	R	–	–
تقدّم بسته			تأخیر	توان خروجی	قابلیّت اطمینان	بلا استفاده	

الف) سه بیت سمت چـپ ، اولویـت بسـتهٔ IP را تعییـن می‌کنـد . اگـر در ایـن سـه بیـت صفر قرار گرفته باشـد بسـته اطلاعـاتی از نـوع معمولـی تلقـی می‌شـود ، یعنـی دارای پـایین تریـن مقـدار اولویـت اسـت و اگـر مقـدار ۷ یعنـی 2(۱۱۱) در ایـن سـه بیـت قـرار گرفتـه باشد بالاترین اولویـت بـرای بسـته درنظـر گرفتـه می‌شـود . مسـیریاب در بیـن بسـته‌های IP کـه از کانالـهای مختلـف وارد شـده‌اند ، بسـته‌هایی را زودتـر پـردازش و مسـیریابی می‌کنـد که دارای حقّ تقدم و اولویـت بـالاتری باشـند. بسـته‌های بـا حـقّ تقـدم بـالا بـرای عملیـاتی نظیر ارسال بسـته‌های اطلاعـاتی بـه منظـور تنظیـم و پیکربنـدی پارامترهـای زیرشـبکه مـورد استفاده قـرار میگـیرد. (مثـلاً بـرای گـزارش یـک خرابـی در زیـر شـبکه یـا مبادلـهٔ جـداول مسـیریابی)

ب) بیتهای **R , T , D** : بیـت D بـه معنـای تـأخیر[2] ، بیـت R بـه معنـای قـابلیت اطمینـان و بیـت T بـه معنـای تـوان خروجـی خط[3] اسـت و ماشـین میزبـان بـا قـرار دادن ۱ در ایـن بیتها انتظارش را از زیرشـبکه بیـان می‌کنـد . مسـیریابها بـا بررسـی ایـن سـه بیـت می‌تواننـد در مـورد انتخـاب مسـیر مناسـب تصمیـم بگـیرند . بعنـوان مثـال یـک کانـال مـاهواره‌ای دارای تـوان خروجـی بسـیار بـالا (از لحـاظ نـرخ ارسـال) ولیکـن تـاخیر نامناسـب اسـت ، در صورتی که یـک خـط اجـاره‌ای می‌توانـد دارای تـاخیر کمـتر و همچنیـن تـوان خروجـی

[1] Reliability
[2] Delay
[3] Throughput

همچنین توان خروجی کمتر باشد . اگر در ارسال یک بستهٔ IP ، تاخیر پذیرفتنی نباشد با یک کردن بیتِ D مسیریاب را وادار می‌کند که حتی الامکان از خطوط پرتاخیر مثل خط ماهواره‌ای استفاده نکند؛ با یک کردن بیت R مسیریاب موظف خواهد بود تا از بین خطوط خروجی امن ترین و کم خطاترین آنها را انتخاب کند. (البته در صورت امکان)

اکثر مسیریابهای تجاری فیلد Type of Service را نادیده می‌گیرند و اهمیتی به محتوای آن نمی‌دهند .

- فیلد **Total Length** : در این فیلدِ ۱۶ بیتی عددی قرار می‌گیرد که طول کل بستهٔ IP را که شامل مجموع اندازهٔ سرآیند و ناحیه داده است ، تعیین می‌کند . مبنای طول برحسب بایت است و بنابراین حداکثر طول کل بستهٔ IP می‌تواند ۶۵۵۳۵ بایت باشد.

- فیلد **Identification** : همانگونه که قبلاً اشاره شد برخی از مواقع مسیریابها یا ماشینهای میزبان مجبورند یک دیتاگرام را به قطعات کوچکتر بشکنند و ماشین مقصد مجبور است آنها را بازسازی کند ، بنابراین وقتی یک دیتاگرامِ واحد شکسته می‌شود باید مشخصه‌ای داشته باشد تا در هنگام بازسازیِ آن در مقصد بتوان قطعه‌های آن دیتاگرام را از بقیه جدا کرد. در این فیلدِ ۱۶ بیتی عَددی قرار می‌گیرد که شمارهٔ یک دیتاگرام واحد را مشخص می‌کند . کلیه بسته‌های IP که با این شماره وارد می‌شوند قطعه‌های مربوط به یک دیتاگرام بوده و باید پس از گردآوری قطعه‌ها ، آن را مجدداً بازسازی کرد. بعنوان مثال اگر در این فیلد عدد ۱۶۵۲ قرار بگیرد تمامی بسته‌های IP که مشخصهٔ ۱۶۵۲ دارند قطعه‌های مربوط به یک دیتاگرام هستند و پس از دریافت کل قطعه‌ها باید بازسازی شوند. البته برای حفظ ترتیب ، هر قطعه گذشته از یک شمارهٔ مشخصه بایستی دارای شمارهٔ ترتیب نیز باشد تا بتوان آنها را طبق این شماره مرتب و بازسازی کرد.

- فیلد **Fragment offset** : این فیلد در سه بخش سازماندهی شده است :

الف) بیت DF[1] : با یک شدن این بیت در یک بستهٔ IP هیچ مسیریابی حقّ ندارد آن را قطعه قطعه کند ، چرا که مقصد قادر به بازسازی دیتاگرام‌های تکه تکه شده نیست . بعنوان مثال وقتی که یک کامپیوتر بدون دیسک از طریق ROM بوت می‌شود اطلاعات هسته اصلی سیستم عامل باید در قالب یک دیتاگرام واحد برای آن کامپیوتر ارسال شود چرا که آن کامپیوتر در حال حاضر نرم افزار لازم برای بازسازی بسته‌های قطعه قطعه شده را ندارد. اگر این بیت به ۱ تنظیم شده باشد و مسیریابی نتواند آنرا به دلیل بزرگی اندازهٔ آن ، انتقال بدهد لاجَرَم آنرا حذف خواهد کرد.

ب) بیت MF[2] : این بیت مشخص می‌کند که آیا بستهٔ IP آخرین قطعه از یک دیتاگرام محسوب می‌شود یا باز هم قطعه‌های بعدی وجود دارد. در آخرین قطعه از یک دیتاگرام بیت MF صفر خواهد بود و در بقیه الزاماً ۱ است.

ج) Fragment offset : این قسمت که سیزده بیتی است در حقیقت شماره ترتیب هر قطعه در یک دیتاگرام شکسته شده محسوب می‌شود. با توجه به سیزده بیتی بودن این فیلد، یک دیتاگرام حداکثر می‌تواند به ۸۱۹۲ تکه تقسیم شود.

نکته بسیار مهم در مورد این فیلد آن است که اندازهٔ هر قطعه باید ضریبی از ۸ باشد یعنی به استثنای قطعهٔ آخر ، اندازه بقیهٔ قطعه‌ها بایستی بگونه ای انتخاب شود که ضریبی از ۸ بایت باشد؛ مثلاً اگر در فیلد آفست مقدار ۷ قرار بگیرد نشان می‌دهد که محل قرار گرفتن قطعهٔ جاری در دیتاگرام بازسازی شده در موقعیت بایت پنجاه وششم (۵۶=۸*۷) خواهد بود. به عنوان مثالی دیگر فرض کنید مسیریابی مجبور است یک دیتاگرام به طول ۵۰۰۰ بایت را قطعه قطعه کند به گونه‌ای که اندازهٔ هر قطعه زیر ۱۵۰۰ بایت باشد. در چنین موردی نمیتواند اندازهٔ هر قطعه را ۱۲۵۰ بایت در نظر بگیرد چرا که ضریبی از ۸ نیست ولی اندازهٔ ۱۲۸۰ مناسب است. در این حالت مسیریاب ، دیتاگرام را به سه بستهٔ ۱۲۸۰ بایتی و یک بستهٔ ۱۱۶۰ بایتی می‌شکند. در این مثال فرض کنید مسیریاب شمارهٔ ۲۳۲۲ را به عنوان مشخصهٔ دیتاگرام انتخاب کرده است؛ بنابراین برای هر یک از چهار قطعهٔ دیتاگرام ، فیلد آفست و مشخصه به صورت زیر خواهد بود:

---

[1] Don’t Fragment
[2] More fragment

شمارهٔ قطعه	Identification	Fragment Offset	بیت MF	آدرس محّل قرار گرفتن قطعه در دیتاگرام	طول هر قطعه
قطعهٔ شماره ۱	2322	0	1	8*0=0	۱۲۸۰
قطعهٔ شماره ۲	2322	160	1	8*160=1280	۱۲۸۰
قطعهٔ شماره ۳	2322	320	1	8*320=2560	۱۲۸۰
آخرین قطعه	2322	480	0	8*480=3840	۱۱۶۰

(بیت اول یعنی بیت سمت چپ از این فیلد که در شکل (۷-۳) به رنگ سیاه علامت گذاری شده مورد استفاده ندارد)

ممکن است یک دیتاگرامِ واحد از یک ماشین میزبان روی زیر شبکه تزریق شود و در طول مسیر به مسیریابی برسد که به دلیلی مجبور به شکستن آن به قطعات کوچکتر شود. در چنین حالتی باز هم وظیفهٔ بازسازی قطعات به عهدهٔ ماشین مقصد می‌باشد. به عبارت ساده تر عمل شکستن یک دیتاگرام در هر جای زیر شبکه ممکن است اتفاق بیفتد ولیکن عمل باز سازی فقط در ماشین مقصد انجام می‌شود.

- فیلد **Time To Live** : این فیلد هشت بیتی در نقش یک شمارنده ، طول عمر بسته را مشخص می‌کند. طول عمر یک بسته بطور ضمنی به زمانی اشاره می‌کند که یک بسته IP می‌تواند بر روی شبکه سرگردان باشد . حداکثر طول عمر یک بسته ، ۲۵۵ خواهد بود که به ازای عبور از هر مسیریاب[1] از مقدار این فیلد یک واحد کم می‌شود. هر گاه یک بستهٔ IP به دلیل بافر شدن در حافظهٔ یک مسیریاب زمانی را معطّل بماند ، به ازای هر ثانیه یک واحد از این فیلد کم خواهد شد. به محض آنکه مقدار این فیلد به صفر برسد بستهٔ IP در هر نقطه از مسیر باشد حذف شده و از ادامهٔ سیرِ آن به سمت مقصد جلوگیری خواهد شد. (البته معمولاً یک پیام هشدار به ماشینی که آن بسته را تولید کرده باز پس فرستاده خواهد شد.)

اگرچه بزرگترین عددی که در فیلد طول عمر بسته قرار می‌گیرد ۲۵۵ است ولی در عمل مقداری که سیستمهای عامل در این فیلد قرار می‌دهند چیزی حدود ۳۰ است. (البته می‌توان مقدار پیش فرض آن را عوض کرد)

این فیلد برای پاکسازی زیر شبکه از بسته‌های IP که به هر دلیل در یک مسیرِ بسته می‌چرخند بسیار حیاتی است وگرنه پس از مدتی کلّ زیر شبکه از بسته‌های

[1] در ادبیات شبکه به عبور بسته از یک مسیریاب یک جهش یا Hop گفته می‌شود.

آشغال پر خواهد شد. بسته‌های سرگردان گاهاً به این دلیل بوجود می‌آیند که جداول مسیریابی در بعضی از مسیریابها آلوده به اطلاعات نادرست[۱] شده‌اند. سرگردانی یک بسته در زیرشبکه مسئلهٔ غیر ممکنی نیست و گاهی اتفاق می‌افتد.

- فیلد **Protocol** : دیتاگرامی که در فیلد داده از یک بستهٔ IP حمل می‌شود با ساختمان دادهٔ خاصّ از لایهٔ بالاتر تحویلِ پروتکل IP شده تا روی شبکه ارسال شود. بعنوان مثال ممکن است این داده‌ها را پروتکل TCP در لایه بالاتر ارسال کرده باشد و یا ممکن است این کار توسط پروتکل UDP انجام شده باشد. بنابراین مقدار این فیلد شمارهٔ پروتکلی است که در لایه بالاتر تقاضای ارسال یک دیتاگرام کرده است؛ بسته‌ها پس از دریافت در مقصد باید به پروتکل تعیین شده تحویل داده شود. فیلد پروتکل ۸ بیتی است و پروتکلهای لایهٔ بالاتر دارای یک شمارهٔ هشت بیتیِ منحصر بفرد و استاندارد هستند که درصورت نیاز به دانستن شمارهٔ آنها می‌توانید به انتهای این فصل مراجعه کنید .

- فیلد **Header Checksum** *:* این فیلد که شانزده بیتی است به منظور کشف خطاهای احتمالی در سرآیند هر بستهٔ IP استفاده می‌شود . برای محاسبهٔ کد کشف خطا ، کلّ سرآیند بصورت دو بایت، دوبایت با یکدیگر جمع می‌شود. نهایتاً حاصل جمع به روش "مکمل یک"[۲] منفی می‌شود و این عدد منفی در این فیلد از سرآیند قرار می‌گیرد .

در هر مسیریاب قبل از پردازش و مسیریابی ابتدا صحّتِ اطلاعات درون سرآیند بررسی می‌شود. روش بررسی بدینصورت است که اگر تمامی سرآیند بصورت دو بایت ، دوبایت در مبنای مکمل یک با یکدیگر جمع شود باید حاصل جمع ، صفر بدست آید؛ در غیر این صورت بستهٔ IP فاقد اعتبار بوده حذف خواهد شد.
دقت کنید که فیلد Checksum در هر مسیریاب باید از نو محاسبه و مقداردهی شود زیرا وقتی یک بستهٔ IP وارد یک مسیریاب می‌شود حداقّل فیلد TTL از آن بسته عوض خواهد شد.

فیلد Checksum برای کشف خطاهای احتمالی درون داده‌های فیلد Payload استفاده نمی‌شود چرا که اینگونه خطاها در لایه پایینتر یعنی لایه فیزیکی معمولاً

---

[۱] Corrupt
[۲] One's Compelement

توسط کدهای CRC نظارت می‌شود؛ در ضمن لایه‌های بالاتر نیز مسئلهٔ خطا را بررسی می‌کنند.

در حقیقت این فیلد برای کشف خطاهایی است که یک مسیریاب در تنظیم سرآیند یک بستهٔ IP مرتکب شده است .

- فیلد **Source Address** : هر ماشین میزبان در شبکهٔ اینترنت یک آدرس جهانی و یکتای ۳۲ بیتی دارد. بنابراین هر ماشین میزبان در هنگام تولید یک بستهٔ IP باید آدرس خودش را در این فیلد قرار بدهد .

بحث آدرسها در اینترنت یکی از مسائل بسیار مهمی است که در فصلی مجزا به آن خواهیم پرداخت . (به این آدرس از این بعد ، "آدرسِ IP" می‌گوئیم)

- فیلد **Destination Address** : در این فیلد آدرس ۳۲ بیتیِ مربوط به ماشین مقصد که باید بسته IP تحویل آن بشود ، قرار می‌گیرد.

- فیلد اختیاری **Options** : در این فیلدِ اختیاری می‌توان تا حداکثر ٤٠ بایت قرار داد و محتوی اطلاعاتی است که می‌تواند به مسیریابها در مورد یافتن مسیر مناسب کمک کند. البته به گونه ای که اشاره شد حداکثر فضای این فیلد ٤٠ بایت است که بسیار کم به نظر می‌رسد.

از آنجایی که در فضای ٤٠ بایتی این فیلد چندین گزینه می‌تواند قرار بگیرد و هر گزینه نیز اندازهٔ متفاوتی دارد (بر حسب بایت) لذا هر گزینه با یک کد یک بایتی مشخص می‌شود:

7	6	5	4	3	2	1	0
Copy Flag	Option Class		Option Number				

- بیت Copy Flag : ۱ بودن این بیت مشخص میکند که اگر مسیریابی مجبور به شکستن بستهٔ فعلی شود ، این گزینه در یکایک قطعات بسته تکرار شود. صفر بودن این بیت به معنای آنست که در هنگام شکسته شدن بسته این گزینه فقط در اولین قطعه وجود داشته باشد.
- دو بیت Option Class : این دو بیت نوع عملکرد گزینه را تعیین میکند:

  00 : عملکرد گزینه

  10 : عملکرد گزینه برای اشکالزدایی و مدیریت شبکه می‌باشد.

01 و 11 : تعریف نشـــده اسـت.

♦ پنـج بیـت Option Number : ایـن پنـج بیـــت نـــوع و معنـــای گزینـــه را مشـــخص میکند. تاکنون پنج گزینهٔ متفـاوت در ایـن فیلـد تعریـف شـده اسـت:

Option Class	Option Number	Name Of Options	شرح
00	0	End of Options List	**۱-تعیین پایـان لیسـت گزینـه‌ها**
00	1	Null Option	**۲-گزینهٔ پوچ (فقط بـرای پـر کـردن فضـا)**
00	2	Security	**۳-گزینهٔ امنیـت**
00	3	Loose Source Routing	**۴-گزینهٔ تعیین مسـیر بصـورت نـاقص**
00	7	Record Route	**۵-گزینهٔ ثبت مسـیر**
00	9	Strict Source Routing	**۶-گزینهٔ تعیین مسـیر بصـورت دقیـق و صریـح**
10	4	Timestamp	**۷-گزینهٔ ثبت مسـیر و زمـان**

گزینهٔ اوّل : با این گزینـه پایـان مجموعـهٔ گزینـه‌ها مشـخص می‌شـود.

گزینـهٔ دوم : ایـن گزینـه هیـچ ارزش اجرایـی نـدارد و فقـط بـرای آنسـت کـه فضـــای فیلد Options به گونه ای پر شـود تـا ضریبـی از ٤ بـاقی بمـاند.

گزینـهٔ سـوم : مشـخص می‌کنـد کـه بسـته IP تـا چـه حـدّ محرمانـه اســت و در ایـــن شـرایط مسـیریاب خواهـد دانسـت کـه ایـن بسـته را از طریـق چـه مسـیرهائی بـــه ســـمت مقصد هدایت نماید تـا امنیـت بسـته تـامین شـود و از چـه مسـیرهائی بـاید احـتراز نمـاید . اکثر مسیریابهای تجـاری از ایـن گزینـه چشمپوشـی می‌نمـایند .

گزینه چـهارم: بـا ایـن گزینـه می‌تـوان مسـیری را بـرای عبـور بسـته (بصـورت نـاقص) مشـخص کـرد و بسـته بـاید قطعـاً از مسـیریابهای مشـخص شـده عبـور نمـاید ولـــی از آن جائی کـه ایـن گزینـه مسـیر کـامل را مشـخص نکـرده اسـت بقیـه مسـیر توسـط مسـیریاب تعیین میشود . بـــرای مثـال فـرض کنیـد بخواهیـد بسـته ای را کـه بـاید از لنـدن بـه سـیدنی طی مسـیر کنـد ، بجـای عبـور از شـرق بـه غـرب از مسـیرهای نیویـورک ، لـس آنجلـس و هانولولو به سـمت سـیدنی ارسـال شـود . کـافی اسـت فقـط آدرس مسـیرهای ابتدائـی را بـا این گزینه مشخص کرده و بقیـه مسـیر بعـهده مسـیریابها گذاشـته شـود.

گزینه پنجم: با درج این گزینه در بستهٔ IP از تمامی مسیریابها خواسته می‌شود که قبل از ارسال بسته به مسیریاب بعدی آدرس خودشان را در فیلد Option ثبت نمایند . با بررسی مسیرهائی که یک بسته از مبداء به سمت مقصد پیموده است می‌توان به اشکالات احتمالی در الگوریتمهای مسیریابی هر مسیریاب پی برد. دقّت کنید که پروتکل IP زمانی وضع شده است که فضای ٤٠ بایتی فیلد Options برای تمامی شبکه‌ها کافی بود؛ چرا که این پروتکل برای اولین بار در ARPANET پیاده شد که حداکثر تعداد مسیریابها در طولانیترین مسیر ، ۹ عدد بود . بنابراین فضای چهل بایتی برای امروزه که هزاران مسیریاب در جهان نصب و راه‌اندازی شده است بسیار ناکافی به نظر می‌رسد.

گزینه ششم : با این گزینه می‌توان مسیر از پیش تعیین شده ای را برای بستهٔ IP تعیین کرد و مسیریابها نیز موظفند از مسیر تعیین شده تبعیت نمایند. با توجه به آنکه در زیرشبکه ، مسیریابی به روشهای پویا انجام می‌شود ، استفاده از این گزینه چندان منطقی و مناسب به نظر نمی‌رسد بلکه فقط بعنوان یک ابزار برای مدیران سیستم جهت آزمایش و بررسی شرایط یک مسیر و تخمین جداول مسیریابی (بصورت دستی) مفید خواهد بود.

گزینه هفتم: این گزینه از تمامی مسیریابها می‌خواهد که زمان دریافت بسته را در فیلد Options درج کنند. این گزینه برای اشکال زدائی از الگوریتمهای مسیریابی مناسب است.

- فیلد **Payload** : در این فیلد داده‌های دریافتی از لایهٔ بالاتر قرار می‌گیرد.

پس از شناسائی ساختار یک بسته IP ، بایستی به مبحث آدرسها در پروتکل IP بپردازیم . مفاهیم آدرسهای IP شما را در درک واقعیت چگونگی مسیریابی بهتر کمک می‌کند . سپس به پروتکلهائی خواهیم پرداخت که به پروتکلIP در لایه شبکه کمک می‌کنند تا یک مسیریابی صحیح امکان پذیر باشد .

۳) مبحث آدرسها در اینترنت و اینترانت

همانگونه که در مباحث قبلی بدان اشاره کردیم پروتکل اینترنت در ارتباطات بین شبکه ای[1] از آدرسهای منحصر به فرد و یکتای ۳۲ بیتی بهره می‌برد. (هر چند که در نسل بعدیِ پروتکل اینترنت که تا سال ۲۰۰۵ همه گیر خواهد شد این آدرسها ۱۲۸ بیتی می‌شوند.) هر ابزار شبکه اعم از ماشینهای میزبان ، مسیریابها و چاپگرهای شبکه در اینترنت با یک آدرس IP شناسائی می‌شوند.

در ادامهٔ این فصل باید موارد زیر را بررسی و مطالعه کنیم:

- قالب هر آدرس IP چگونه سازماندهی می‌شود؟
- کلاسهای مختلف آدرسهای IP به چه منظور و چگونه سازماندهی می‌شوند؟
- چگونه آدرسهای IP به آدرسهای سخت افزاری لایه فیزیکی تبدیل خواهد شد و قراردادهای نمایش آدرسهای IP چگونه هستند؟
- یک مسیریاب چگونه می‌تواند از یک آدرس چهاربایتی ، محل دقیق یک ماشین را بین دهها میلیون ماشین متصّل به شبکه پیدا نماید؟

آدرسهای IP درون یک عدد دودویی ۳۲ بیتی درج می‌شوند ولیکن برای سادگی نمایش به چهار بایت تقسیم شده و بصورت چهار عدد دهدهی که با نقطه از هم جدا شده‌اند نوشته می‌شود؛ یعنی معادل دهدهی هر یک از بایتهای آدرس بصورت مجزا نوشته شده و هر عدد با یک علامتِ . از دیگری تفکیک می‌شود. بعنوان مثال آدرس زیر یک آدرس IP معتبر می‌باشد که در قالب چهار قسمت دهدهی نوشته شده است:

34.21.225.1

این آدرس بصورت زیر در فیلد آدرس از یک بستهٔ IP تنظیم میشود:

پرارزشترین بایت یعنی اولین بایتِ سمت چپ از آدرس IP ، کلاس آدرس را مشخص می‌کند و از این رو دارای اهمیت ویژه است. ولی قبل از آنکه کلاسهای آدرس را تشریح نماییم بازهم روی این نکته تکیه می‌کنیم که وقتی یک ماشین میزبان به شبکه اینترنت متصل می‌شود بایستی آدرس IP آن منحصر به فرد و یکتا[2] باشد . در

---

[1] Internetworking
[2] Unique

در حقیقت هر ماشین روی شبکه با یک آدرس یکتا هویت پیدا میکند . برای اطمینان از یکتا بودن آدرسهای IP برای ارتباطات عمومی، مرکز InterNIC[1] کنترل و نظارت بر روی آدرسهای IP را بر عهده گرفته است.

IANA[2] قدرت اجرائی برای اختصاص آدرسهای IP منحصر به فرد را فراهم کرده است . هر چند شبکه‌های خصوصی که به اینترنت وصل نیستند می‌توانند از آدرسهای IP دلخواه استفاده کنند ولی اگر این شبکه‌ها زمانی بخواهند به اینترنت وصل شوند دوگانگی آدرسهای غیر یکتا و نهایتاً تناقض و اشکال در مسیریابی[3] رخ خواهد داد؛ به همین دلیل پیشنهاد شده است که حتی شبکه‌های خصوصی نیز برای اختصاص آدرس به ماشینهای میزبان از مرکز InterNIC مجوز بگیرند و از آدرسهای معتبر و اختصاصی استفاده کنند.

۱-۳) کلاسهای آدرسِ IP

از آنجا که TCP/IP برای شبکه‌های با مقیاس بزرگ طراحی شده است لذا نمی‌توان انتظار داشت که فضای ۳۲ بیتی آدرس که حدود چهار میلیارد و سیصد میلیون (4,294,967,295) آدرس را در اختیار می‌گذارد ، بدون هیچ نظم و سیاق خاص به ماشینهای شبکه اختصاص داده شود . این کار همانند آن خواهد بود که تمامی آپارتمانها و منازل در کل جهان با شماره‌های ده رقمی مشخص شود بدون آنکه هیچ ضابطه ای در شماره گذاری آنها رعایت شده باشد . آنگاه منزلی با شماره ۱۰٦٥٤۳۲۳۹۰ چگونه پیدا می‌شود !

آدرسهای پستی ساختاری سلسله مراتبی به صورت زیر دارند، به گونه ای که هر منزل در هر کجای دنیا قابل آدرس دهی است و به راحتی پیدا می‌شود:

شماره/کوچه/خیابان/ناحیه/شهر/کشور

فلسفهٔ کلاسهای آدرس IP به همین منظور است :

آدرس ماشین/آدرس زیرشبکه/آدرس شبکه

---

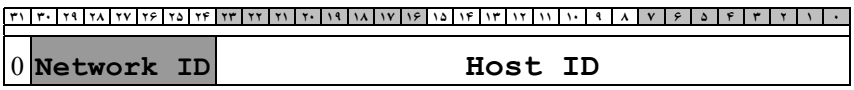
[1] Internet Network Information Center
[2] Internet Assigned Number Authority
[3] Conflict

با توجه به آنکه اینترنت مجموعه‌ای از شبکه‌های متصل شده به هم است برای آدرس دادن به ماشینهای میزبان بهتر است ۳۲ بیت آدرسِ IP به قسمتهای زیر تقسیم شود:

الف) آدرس شبکه
ب) آدرس زیر شبکه (درصورت لزوم)
ج) آدرس ماشین میزبان

آدرسهای IP درپنج کلاس E,D,C,B,A معرفی شده‌اند که شما بایستی آنها را بدقت بشناسید و تحلیل کنید. در زیر قالب کلاسهای پنج گانهٔ آدرس IP مشخص شده است:

- **آدرسهای کلاس A** : قالب ۳۲ بیتیِ آدرس در کلاس A به صورت زیر است:



در کلاس A ، پرارزشترین بیت از آدرس ، مقدار صفر دارد و این بیت کلاس A را از دیگر کلاسها متمایز می‌کند؛ ۷ بیت بعدی "مشخصهٔ آدرس شبکه" و سه بایت باقیمانده ، آدرس ماشین میزبان را تعیین می‌کند. بنابراین در کلاس A بایت پرارزش در محدوده صفر تا ۱۲۷ تغییر می‌کند. چون با ۲٤ بیت می‌توان حدود هفده میلیون ماشین میزبان را آدرس دهی کرد ، می‌توان به این نتیجه رسید که آدرسهای کلاس A بایستی برای آژانسهای ستون فقرات اینترنت یا شبکه‌ها بسیار عظیم مثل NSFNet یا ARPANet اختصاص داده شده باشد . مشخصهٔ شبکه در این کلاس بهیچوجه نمی‌تواند اعداد صفر یا ۱۲۷ انتخاب شود چرا که این دو عدد در شبکه معنای دیگری خواهند داشت و بعداً به آن اشاره خواهیم کرد. بنابراین تعداد شبکه‌هائی که در جهان می‌توانند از کلاس A استفاده کنند ۱۲٦ تا خواهد شد که بسیار کم است . امروزه اختصاص آدرسهای کلاس A غیر ممکن است چرا که همهٔ آنها توسط پیشگامان شبکه سالها قبل تملیک شده‌اند.

وقتی به یک آدرس IP که در قالب دهدهی نوشته شده است نگاه می‌کنید براحتی می‌توانید کلاس آنرا تشخیص بدهید . اگر عدد سمت چپ آدرس ، بین صفر تا ۱۲۷ باشد ، آن آدرس از کلاس A خواهد بود:

74. 103 .14 .138

Net ID Host ID

آدرس IP (127.0.0.0) در پروتکل اینترنت ، یک شبکه را تعیین نمی‌کند بلکه بصورت قراردادی بعنوان آدرس "حلقه بازگشت"[1] جهت اهداف اشکال زدایی استفاده شده است چرا که این آدرس عملاً معادل آدرس خود ماشین محلی است .

- **آدرسهای کلاس B**: قالب ۳۲ بیتیِ آدرس در کلاس **B** به صورت زیر است:

هر گاه دو بیت پرارزش از آدرس IP مقدار 10 داشته باشد آن آدرس از کلاس B خواهد بود . ١٤ بیت باقیمانده از ۲ بایت سمت چپ ، آدرس شبکه را تعیین می‌کند و دو بایت اول از سمت راست (١٦ بیت) آدرس ماشین میزبان خواهد بود .

در آدرسهای کلاس B ، تعداد ١٦٣٨٢ (2^{14}-2) شبکهٔ گوناگون قابل تعریف خواهد بود و هر شبکه می‌تواند ٦٥٥٣٤ (2^{16}-2) ماشین میزبان تعریف نماید. اختصاص آدرسهای کلاس B برای شبکه‌های بسیار عظیم مناسب است . هر چند تعداد این شبکه در جهان می‌تواند تا حدود شانزده هزار عدد باشد ولیکن امروزه عملاً نمی‌توان آدرس کلاس B گرفت چرا که تقریباً همهٔ آنها آن تخصیص داده شده‌اند .

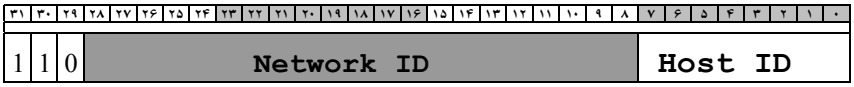
اگر آدرس IP به صورت دهدهی نوشته شود و عدد سمت چپِ آن بین ۱۲۸ تا ۱۹۱ باشد ، آن آدرس ، کلاس B خواهد بود:

134. 64. 143 . 24

Net ID Host ID

[1] Loopback

➢ **آدرس کلاس C** : قالب ۳۲ بیتیِ آدرس در کلاس **C** به صورت زیر است:



کلاس C مناسب ترین و پرکاربردترین کلاس از آدرس‌های IP است . همانگونه که از شکل مشخص است در این کلاس ، سه بیت پرارزش دارای مقدار 110 است و ۲۱ بیت بعدی از سه بایت سمت چپ برای تعیین آدرس شبکهٔ مورد نظر بکار رفته است . بنابراین در این کلاس می‌توان حدود دو میلیون شبکه را در جهان آدرس دهی کرد و هر شبکه می‌تواند تا ۲۵٤ عدد ماشین میزبان تعریف نماید . برای تشخیص آدرس‌های کلاس C به عدد سمت چپ از آدرس IP که به صورت دهدهی نوشته شده است نگاه کنید. اگر عدد بین ۱۹۲ تا ۲۲۳ بود آن آدرس از کلاس C خواهد بود:

199.164. 78 .132

Net ID Host ID

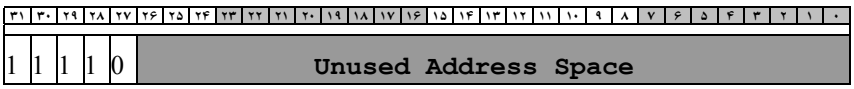
➢ **آدرس کلاس D** : قالب ۳۲ بیتیِ آدرس در کلاس **D** به صورت زیر است:

در این کلاس ، چهار بیت پرارزش دارای مقدار 1110 است و ۲۸ بیت باقیمانده از کلّ آدرس برای تعیین آدرسهای "چند مقصده"[1] (آدرسهای گروهی) است .

از این آدرسها برای ارسال یک دیتاگرام به طور همزمان برای چندین ماشین میزبان کاربرد دارد و بمنظور عملیات رسانه ای و چند پخشی بکار می‌رود. توضیح بیشتر در مورد این کلاس در بخشی مجزا ارائه خواهد شد .

[1] Multicast

➢ **آدرس کلاس E** : قالب ۳۲ بیتیِ آدرس در کلاس **E** به صورت زیر است:



فعلاً این دسته از آدرسها که پنج بیت پرارزش آنها در سمت چپ 11110 است کاربرد خاصّی ندارند و برای استفاده در آینده بدون استفاده رها شده‌اند. البته گاهی بصورت آزمایشی از این آدرسها استفاده شد ولی تاکنون جهانی نشده‌اند .

۲-۳) آدرسهای خاصّ

در بین تمامی کلاسهای آدرسIP پنج گروه از آدرسها ، معنای ویژه ای دارند و با آنها نمی‌توان یک شبکهٔ خاص را تعریف و آدرس دهی کرد. این پنج گروه آدرس عبارتند از:

الف) آدرس 0.0.0.0 : هر ماشین میزبان که از آدرس IP خودش مطلع نیست این آدرس را بعنوان آدرس خودش فرض می‌کند. البته از این آدرس فقط به عنوان آدرس مبداء و برای ارسال یک بسته می‌توان استفاده کرد و گیرندهٔ بسته نمی‌تواند پاسخی به مبداء بسته برگرداند.[۱]

ب) آدرس 0.HostID زمانی به کار می‌رود که ماشین میزبان ، آدرس مشخصهٔ شبکه ای که بدان متعلق است را نداند. در این حالت در قسمت NetID مقدار صفر و در قسمت HostID شمارهٔ مشخصهٔ خود را قرار می‌دهد.

ج) آدرس 255.255.255.255 : برای ارسال پیامهای فراگیر برای تمامی ماشینهای میزبان بر روی شبکهٔ محلی که ماشین ارسال کننده به آن متعلق است .

د) آدرس NetID.255 : برای ارسال پیامهای فراگیر برای تمامی ماشینهای یک شبکهٔ راه دور که ماشین میزبان فعلی متعلق به آن نیست. آدرس شبکهٔ مورد نظر در قسمت NetID تعیین شده و تمامی بیتهای قسمت مشخصهٔ ماشین میزبان ۱ قرار داده می‌شود. البته بسیاری از مسیریابها برای مصون ماندن شبکه از مزاحمتهای بیرونی ، چنین بسته‌هایی را حذف می‌کنند.

---

[۱] **استفاده از این آدرس مانند آنست که در آدرس فرستندهٔ یک بستهٔ پستی نوشته شود : "خودم" . بسته می‌تواند به مقصد برسد ولی پاسخی نخواهد داشت.**

ه) 127.xx.yy.zz بعنوان "آدرس بازگشت" شناخته می‌شود و آدرس بسیار مفیدی برای اشکالزدایی از نرم افزار می‌باشد. به عنوان مثال اگر بسته ای به آدرس 127.0.0.1 ارسال شود ، بسته برای ماشین تولیدکنندهٔ آن بر خواهد گشت[1]؛ در این حالت اگر نرم افزارهای TCP/IP درست و بدون اشکال نصب شده باشد فرستنده بسته باید آنرا مجدداً دیافت کند. همچنین از این آدرس می‌توان برای آزمایش برنامه‌های تحت شبکه ، قبل از نصب آنها بر روی ماشینهای میزبان استفاده کرد.

۳-۳) آدرسهای زیرشبکه

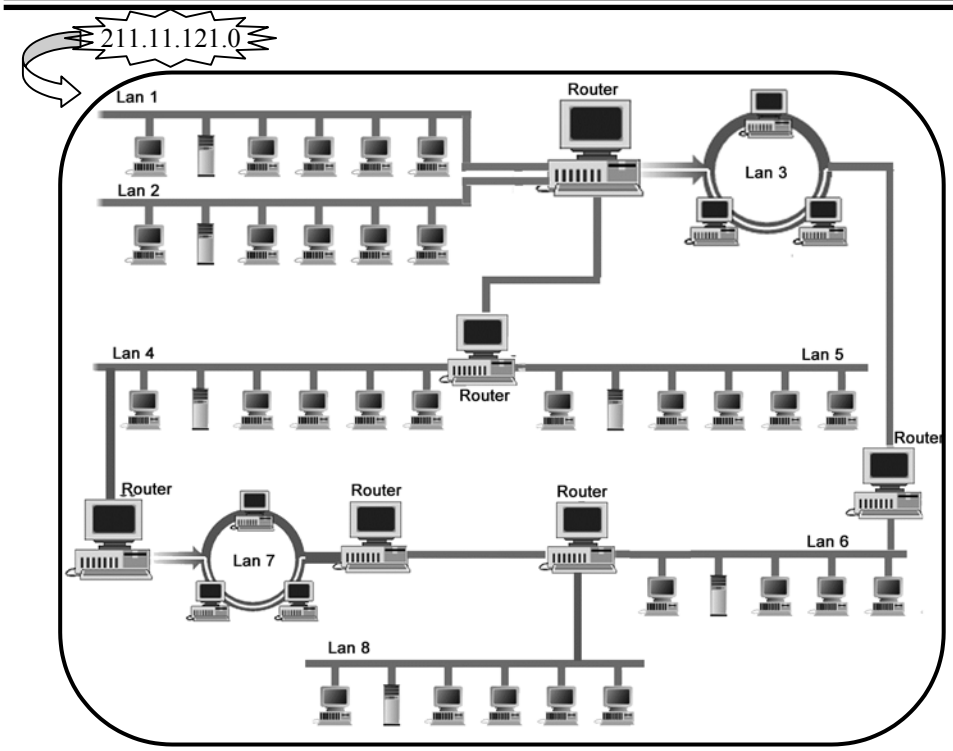
در ادامهٔ بحث بایستی مسئله زیر شبکه را در خصوص آدرس دهی‌ها مطرح نمائیم . مبحث را با یک مثال آغاز می‌نمائیم:

فرض کنید دانشگاه شما یک کلاس C با قابلیت تعریف ۲۵٤ ماشین میزبان ثبت می‌نماید (مثلاً 211.11.121.0) ؛ یعنی شبکهٔ دانشگاه توانایی آدرس دهی ۲۵۵ ایستگاه را در شبکه دارد . در نظر بگیرید که دانشگاه دارای یک شبکهٔ محلی واحد و یکپارچه برای کل دانشگاه نیست بلکه دارای هشت شبکهٔ محلی مجزا است که برای هر دانشکده تهیه دیده شده است؛ (همانند ساختاری که در شکل (۸-۳) ترسیم شده است). هر کدام از این شبکه‌ها که می‌تواند توپولوژی متفاوتی داشته باشد ، از طریق مسیریاب به هم متصل شده‌اند و طبعاً برای ارتباط بین شبکه‌های هر دانشکده باید مسیریابی صورت گیرد. از دیدگاه بیرونی کلّ مجموعهٔ شبکه‌های محلی دانشگاه با یک آدرس مشخصه یعنی 211.11.121.0 شناخته میشود و مسیریابهای بیرونی هیچ شناختی از ساختار شبکه بندی داخلی دانشگاه ندارند. (هر یک از شبکه‌های محلی داخل دانشگاه یک زیرشبکه نامیده می‌شود.) بنابراین باید روشی وجود داشته باشد تا از طریق آدرسهای کلاس C (یا هر کلاس دیگر) بتوان زیر شبکه‌ها را نیز مشخص کرد تا مسیریابهای داخلی نیز قادر باشند زیر شبکه‌های مختلف را شناسایی و تفکیک کنند.

این مسئله برای آدرسهای کلاس B و A بسیار ضروری و اجتناب ناپذیر می‌نماید چرا که نمی‌توان انتظار داشت که یک موسسه که آدرس کلاس B با قابلیت تعریف حدود ٦٦ هزار ماشین میزبان ثبت کرده است فقط یک شبکهٔ یکپارچه داشته باشد بلکه چنین موسسه ای ممکن است دارای صدها زیر شبکهٔ کوچک و بزرگ باشد.

---

[1] **این آدرس همانند آنست که فرستندهٔ یک بستهٔ پستی آدرس دقیق خودش را به عنوان گیرندهٔ آن درج نماید. بنابراین با آدرس 0.0.0.0 تفاوت ذاتی دارد.**



شکل (۸-۳) یک شبکهٔ خود مختار که کلاً با یک آدرس مشخصهٔ شبکه شناسایی میشود.

برای آنکه بتوان زیرشبکه‌ها[۱] را تفکیک کرد جدای از قسمت آدرس شبکه که کل شبکهٔ دانشگاه شما را مشخص می‌کند بایستی در قسمت مشخصهٔ ماشین میزبان نیز به گونه ای زیر شبکه‌ها مشخص شوند . این کار از طریق مفهومی به نام "الگوی زیرشبکه"[۲] انجام میشود.

شما با نگاه اول به اولین عدد سمت چپ متوجه خواهید شد که این آدرس از چه کلاسی است ولی هنوز موارد مبهمی وجود دارد : آیا شبکه ای که آدرس آنرا پیش رو دارید فقط یک شبکه است یا خودش زیر شبکه بندی شده است؛ یعنی از چند شبکهٔ محلی متصل بهم تشکیل شده است؟

---

[۱] Subnetworks
[۲] Subnet Mask

این اطلاعات برای شبکه‌های مبتنی بر TCP/IP که قابلیت مسیریابی دارند بسیار مهم است ، چرا که هر ماشینِ میزبان بایستی قادر به درک این مطلب باشد که آیا یک ماشین مقصد با آدرس خاصّ و مشخص ، بر روی شبکهٔ محلی خودش واقع است یا آنکه آن آدرس متعلق به زیر شبکه دیگری است . بر اساس این اطلاعات ماشین میزبان تصمیم می‌گیرد که آیا انتقال اطلاعات باید مستقیماً بر روی شبکهٔ محلی انجام شود یا آنکه باید از طریق یک مسیریاب روی شبکه ای دیگر ارسال شود .

تمامی ماشینهای میزبان برای تشخیص محل مقصد یک بستهٔ IP در شبکه احتیاج به یک مشخصهٔ دیگر دارند و آن "الگوی زیرشبکه" نامیده می‌شود.

الگوی زیرشبکه یک عدد ۳۲ بیتی دودویی است که برای ماشین میزبان نقش یک مقایسه‌گر را بازی می‌کند تا با استفاده از آن بتواند تشخیص دهد که آیا مقصد روی همین شبکه محلی است که خودش به آن تعلق دارد یا روی شبکه دیگری است .

فرآیند استفاده از "الگوی زیرشبکه" را با استفاده از مثال قبل ولی با آدرس کلاس B آموزش می‌دهیم:

فرض کنید شما کاربری روی یک ایستگاه در شبکه دانشگاه خودتان هستید, آدرس IP متعلق به دستگاه شما بصورت زیر اختصاص داده شده است :

131.55.213.73

با یک نگاه متوجه می‌شوید که آدرس از کلاس B است که مشخصهٔ شبکهٔ آن معادل 131.55.0.0 و مشخصهٔ ماشین شما 0.0.213.73 است؛ ولی هنوز نمی‌دانید شبکه ای که مشخصهٔ آن معادل 131.55 است آیا زیر شبکه دارد یا خیر؟

فرض کنید که دانشگاه شما با آدرس شبکه 131.55.0.0 ، می‌خواهد حداکثر دارای ۲۵٤ زیر شبکه باشد ، بهمین دلیل فرض کرده است که در فیلد مشخصهٔ ماشین میزبان (Host ID) که در کلاس B دو بایت سمت راست را شامل میشود ، بایت دومِ آن به عنوان مشخصهٔ مربوط به زیر شبکه تعریف شود. یعنی فیلد دوبایتی مربوط به مشخصهٔ ماشین میزبان به دو بخش تقسیم شده است:

الف) مشخصهٔ زیرشبکه ب) مشخصهٔ ماشین میزبان

۳۱	۳۰	۲۹	۲۸	۲۷	۲۶	۲۵	۲۴	۲۳	۲۲	۲۱	۲۰	۱۹	۱۸	۱۷	۱۶	۱۵	۱۴	۱۳	۱۲	۱۱	۱۰	۹	۸	۷	۶	۵	۴	۳	۲	۱	۰
1	0	Network ID																						Subnet ID						Host ID	

ماشین شما تصمیم دارد بسته ای را برای یک ماشین میزبان با آدرس IP معادل 131.55.108.75 بفرستد؛ ماشین از کجا می‌تواند بفهمد که مقصد روی همین شبکه محلی که شما بدان متعلق هستید واقع است یا آنکه به شبکهٔ محلی در یک دانشکدهٔ دیگر متعلق است . دانستن این موضوع بسیار با اهمیت خواهد بود چرا که اگر ماشین میزبان مورد نظر روی شبکهٔ دیگری باشد بسته باید با آدرس فیزیکی "مسیریاب پیش فرض"[1] روی کانال ارسال شود. بنابراین تمام ماشینهای روی شبکه بایستی از وضعیت زیر شبکه‌ها مطلع باشند .

با توجه به آنچه که در بالا اشاره شد دومین بایت از سمت راست بعنوان مشخصهٔ زیر شبکه اختصاص داده شده است و بهمین دلیل هر ماشین برای دانستن آنکه آیا ماشین مقصد در شبکه محلی خودش واقع است یا در خارج از شبکه قرار دارد باید قسمت "مشخصهٔ شبکه" و "مشخصهٔ زیرشبکه" از آدرس IP خودش را با همین مشخصه‌ها از آدرس مقصد مقایسه نماید.

اینجاست که یک الگوی ۳۲ بیتی تعریف می‌شود که یک عدد ۳۲ بیتی و در این مثال بصورت 255.255.255.0 است:

۳۱	۳۰	۲۹	۲۸	۲۷	۲۶	۲۵	۲۴	۲۳	۲۲	۲۱	۲۰	۱۹	۱۸	۱۷	۱۶	۱۵	۱۴	۱۳	۱۲	۱۱	۱۰	۹	۸	۷	۶	۵	۴	۳	۲	۱	۰
۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۰	۰	۰	۰	۰	۰	۰	۰

هر گاه ماشین بخواهدیک آدرس IP را تحلیل کند . الگوی فوق را با آدرس IP خودش AND می‌کند. (با اینکار در حقیقت Host ID خودش را صفر می‌نماید) سپس مجدداً الگو را با آدرس IP مقصد AND می‌کند (مشخصهٔ ماشین مقصد هم صفر می‌شود) حال نتیجه دو مرحله را با هم مقایسه می‌نماید . اگر نتیجه دو مرحله یکسان بود ، هم مشخصهٔ شبکه و هم مشخصهٔ زیرشبکه از آدرسهای مبدأ و مقصد یکی است و هر دو روی یک شبکهٔ محلی قرار دارند. در صورت عدم تساوی ، ماشین مبدأ به این نتیجه می‌رسد که مقصد مورد نظر روی شبکه محلی خودش نیست و آن بسته بایستی به آدرس فیزیکی مسیریابِ پیش فرض ارسال شود.

فرض کنید بسته ای با آدرسهای مشخص زیر بخواهد ارسال شود:

آدرس ماشین مبدأ: 131.55.213.73

[1] Default Gateway

آدرس ماشین مقصد: 131.55.108.75

الگوی زیرشبکه: 255.255.255.0

آدرس ماشین مبدأ در قالب دودویی

AND

الگوی زیرشبکه در قالب دودویی

حاصل مرحلهٔ ۱ :

آدرس ماشین مقصد در قالب دودویی

AND

الگوی زیرشبکه در قالب دودویی

حاصل مرحلهٔ ۲ :

حاصل مراحل ۱ و ۲ با هم مساوی نیستند و بنابراین ماشین مبدأ متوجه خواهد شد که ماشین مقصد روی شبکه محلی خودش نیست و بسته اطلاعاتی را بایستی به آدرس فیزیکی مسیریاب پیش فرض ارسال نماید.

به عنوان مثالی دیگر فرض کنید ماشین شما می‌خواهد برای ماشینی با آدرس IP زیر بسته ای را ارسال نماید:

131.55.213.84

آدرس ماشین مبدأ در قالب دودویی

AND

الگوی زیرشبکه در قالب دودویی

حاصل مرحلهٔ ۱ :

آدرس ماشین مقصد در قالب دودویی

AND

الگوی زیرشبکه در قالب دودویی

حاصل مرحلهٔ ۲ :

همانگونه که مشاهده میشود حاصل مراحل ۱ و ۲ مساوی هستند و بالطبع مبدأ و مقصد روی یک شبکه محلی واقعند و هیچ لزومی ندارد که ارسال بسته به آدرس فیزیکی مسیریاب پیش فرض انجام شود ، بلکه باید مستقیماً از آدرس فیزیکی ایستگاه مقصد استفاده شود .

ذکر این نکته ضروری است که الگوی زیرشبکه باید به عنوان یکی از پارامترهای پیکربندی TCP/IP تنظیم شود و فقط برای تشخیص محل شبکهٔ مقصد کاربرد دارد.

الگوی زیرشبکه در مثالهای بالا ساده ترین حالت بود که به آنها "الگوی زیرشبکهٔ استاندارد"[1] گفته می‌شود چرا که الگوها دقیقاً هشت بیتی هستند .

[1] Standard Subnet Mask

۴) زیر شبکه‌های غیر استاندارد

الگوهای زیرشبکه برای تقسیم فضای آدرس‌دهی در شبکه‌های کلاس A ، B و C به تعدادی زیرشبکه ، تعریف می‌شوند. در مثالهائی که بررسی کردیم الگوی شبکه بصورت زیر تعریف شده بود:

255.255.255.0

حال الگوی زیر را در نظر بگیرید:

255.255.240.0

عدد ۲٤۰ در الگوی زیرشبکه چه چیزی را تعریف می‌کند؟
به فرم دودویی الگوی بالا دقت کنید:

۳۱	۳۰	۲۹	۲۸	۲۷	۲۶	۲۵	۲۴	۲۳	۲۲	۲۱	۲۰	۱۹	۱۸	۱۷	۱۶	۱۵	۱۴	۱۳	۱۲	۱۱	۱۰	۹	۸	۷	۶	۵	۴	۳	۲	۱	۰
۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰

بفرض اگر الگوی بالا برای زیر شبکه بندی آدرس کلاس B به کار رفته باشد ، نشان دهنده آن است که چهار بیت پرارزش از بایت دوم برای تعیین شماره زیر شبکه به کار رفته است و ۱۲ بیت باقیمانده بعنوان "مشخصهٔ ماشین میزبان" استفاده شده است؛ بدین معنا که با این الگو می‌توان ١٤ زیر شبکه (2^4-2) تعریف کرد بگونه ای که در هر زیرشبکه ٤۰۹٤ ، (2^{12}-2) ماشینِ میزبان قابل آدرس دهی خواهد بود.

فراموش نکنید که همیشه تعداد زیرشبکه‌ها و ماشینهای میزبان از کلّ تعداد قابل تعریف ، دوتا کمتر است؛ چراکه زیرشبکه یا ماشینی که تمام بیتهای آن صفر یا تماماً یک باشد قابل تعریف نیست.

کلاً برای آنکه با تنظیم الگوهای زیر شبکه آشنا شوید الگوریتم آنرا در زیر تشریح می‌کنیم:
الف) با دقت و دوراندیشی کافی تعیین کنید چه تعداد زیر شبکه و چه تعداد ماشین میزبان روی هر زیرشبکه خواهید داشت. به تعداد زیرشبکه‌ها و ماشینها عدد ۲ را اضافه کرده و سپس تعیین کنید هر کدام از این اعداد به حداقل چند بیت نیاز دارند.
ب) الگوی ۳۲ بیتی زیرشبکه را بگونه ای تنظیم کنید که در سمت راست آن به تعداد بیتی که برای آدرس دهی ماشینهای میزبان نیاز است صفر قرار بگیرد . بیتهای باقیمانده را هم تماماً ۱ قرار بدهید.
ج) الگوی دودویی را بفرم دهدهی نقطه دار تبدیل کنید.
د) زیرشبکه‌ها و ماشینهای میزبان روی هر زیرشبکه را تعریف نمائید.

حال با مثالی این روند را بهتر بررسی می‌کنیم:

فرض کنید یک شرکت بزرگ دارای حداکثر ۲۵ شبکه محلی است که هر شبکه محلی حداکثر تا ۱۰۰۰ میزبان را حمایت می‌نماید . الگوی زیر شبکه را تنظیم نمائید.

الف) برای آدرس دهیِ ۲۵ زیرشبکهٔ محلی ۵ بیت کفایت می‌نماید . (با ۵ بیت می‌توان سی زیرشبکه (2-32) را تعریف کرد)

ب) آدرس از کلاس B است پس قسمت ۱۶ بیتی از مشخصهٔ ماشین میزبان ، پس از کسر ۵ بیت که بعنوان زیر شبکه استفاده شد مقدار ۱۱ بیت خواهد بود. حداکثر تعداد ماشین میزبان قابل تعریف بصورت زیر حساب می‌شود:

$$2^{11}-2=2046$$

تعداد ماشین قابل تعریف از تعدادی که نیاز داشتیم خیلی بیشتر است که اجازه می‌دهد زیرشبکه در آینده گسترش یابد . پس الگوی زیرشبکه برای شبکهٔ این شرکت به صورت زیر تنظیم می‌شود:

۳۱	۳۰	۲۹	۲۸	۲۷	۲۶	۲۵	۲۴	۲۳	۲۲	۲۱	۲۰	۱۹	۱۸	۱۷	۱۶	۱۵	۱۴	۱۳	۱۲	۱۱	۱۰	۹	۸	۷	۶	۵	۴	۳	۲	۱	۰
۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۱	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰	۰

الگوی زیرشبکه 255.255.248.0

دقت کنید که اگر دور اندیشی کافی نداشته باشید و نیاز باشد که زمانی ساختار زیر شبکه را توسعه بدهید، برای تغییر یک الگوی زیر شبکه به الگوی دیگر ، باید پیکربندی[۱] تمام ماشینهای شبکه بصورت دستی تنظیم شود که دردسر زیادی دارد. لذا از میزان رشد آدرسهای زیرشبکهٔ خود اطمینان حاصل کنید و طرح را مطابق با نیازتان بریزید. هرگونه اشتباه در تنظیم الگوی زیرشبکه منجر به عدم کارکرد صحیح شبکه خواهد شد.

بگونه ای که اشاره شد وقتی تعداد زیر شبکه‌ها را حساب می‌نمائید نهایتاً عدد ۲ را با آن جمع کنید و سپس تعداد بیتهای مورد نیاز را برای عدد بدست آمده محاسبه نمائید. مثلاً فرض کنید که کل شبکه دقیقاً چهار زیرشبکه داشته باشد؛ پس برای الگوی زیرشبکه باید سه بیت اختصاص بدهید.

[۱] Configuration

۵) پروتکل ICMP[۱]

پروتکل IP ، پروتکلی "بدون اتصّال"[۲] و "غیر قابل اعتماد" [۳] است! بدون اتصّال بدین معنا که مسیریاب هر بسته را بدون هیچگونه هماهنگی با مقصد بسته یا مسیریاب بعدی ارسال می‌نماید ، بدون آنکه بتواند اطلاعی از وجود یا عدم وجود مقصد داشته باشد. در ضمن هر مسیریاب پس از ارسال یک بسته آنرا فراموش می‌کند و منتظر "پیام دریافت بسته"[٤] از گیرندهٔ آن نخواهد ماند. اگر یک بستهٔ IP با خطا به مقصد برسد و یا اصلا به مقصد نرسد این پروتکل هیچ اطلاعی در مورد سرنوشت آن به فرستندهٔ بسته نمی‌دهد.

دلایل مختلفی برای نرسیدن یک بسته به مقصد وجود دارد: ممکن است "زمان حیات[٥]" بسته قبل از رسیدن به مقصد منقضی شود؛ ممکن است مسیر یاب بسته را به مسیری اشتباه هدایت کند؛ ممکن است در هنگام قطعه قطعه کردن بسته و ارسال آنها ، یکی از قطعات دچار خطا شود یا به هر دلیلی به مقصد نرسد بنابراین کل دیتاگرام قابل بازسازی نخواهد بود؛ ممکن است مقصدِ بسته آمادگی دریافت بسته را نداشته باشد یا اصلاً وجود خارجی نداشته باشد. در هنگام بروز هرگونه خطا ، پروتکل IP به فرستندهٔ بسته هیچ اطلاعی در مورد سرنوشت آن نخواهد داد .

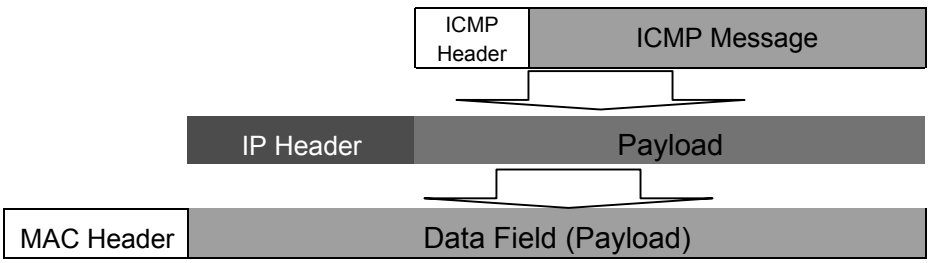
عدم گزارشِ خطا به تولید کنندهٔ یک بسته منجر به تکرار خطا و حمل بیهوده و زائد بسته‌هایی میشود که محکوم به فنا و حذف در شبکه هستند. به عنوان مثال عدم گزارش در مورد آماده نبودن مقصد برای دریافت بسته باعث خواهد شد که فرستندهٔ آن اقدام به ارسال بسته‌های دیگر کند در حالی که این کار بی ثمر خواهد بود و فقط بار ترافیک شبکه را افزایش می‌دهد و حتی می‌تواند منجر به بروز "ازدحام"[٦] شود.

پروتکلICMP در کنار پروتکل IP ، برای بررسی انواع خطا و ارسال پیام برای مبدأ بسته در هنگام بروز اشکالات ناخواسته استفاده می‌شود. در حقیقت ICMP یک سیستم گزارش خطا است که بر روی پروتکلIP نصب می‌شود تا در صورت بروز هرگونه خطا به فرستندهٔ بسته پیام مناسب را بدهد تا آن خطا تکرار نشود. در واقع ICMP وظیفه ای در قبال وقوع خطا ندارد بلکه فقط پیامی که بیانگر بروز خطا و نوع آن است به فرستنده برمیگرداند . این پروتکل اشکالات موجود را در قالب یکسری پیام گزارش می‌کند که این پیام خود در یک بسته IP قرار

---

[۱] Internet Control Message Protocol
[۲] Connectionless
[۳] Unreliable
[٤] Acknowledgement Message
[٥] Time To Live
[٦] Congestion

می‌گیرد که از جانب یک مسیریاب یا ماشین مقصد به آدرس فرستنده باز می‌گردد. در شکل (۹-۳) چگونگی قرار گرفتن یک پیام ICMP درون یک بستهٔ IP تصویر شده است.



شکل (۹-۳) چگونگی قرار گرفتن یک پیام ICMP درون یک بستهٔ IP

با توجه به آنکه پیام ICMP خود درون یک بسته IP جاسازی می‌شود بنابراین فیلد Protocol در سرآیند بستهٔ IP باید با شمارهٔ مشخصهٔ پروتکل ICMP (یعنی ۱) تنظیم شود.

دقّت کنید که خود بسته‌های ICMP نیز ممکن است دچار خطا شوند که برای این گونه خطا پیامی ارسال نخواهد شد .

شکل کلّی و قالب پیام ICMP در زیر مشخص شده است:

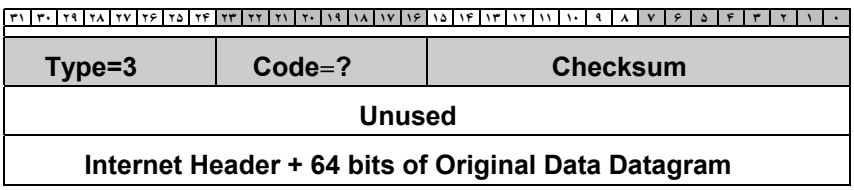
۳۱	۳۰	۲۹	۲۸	۲۷	۲۶	۲۵	۲۴	۲۳	۲۲	۲۱	۲۰	۱۹	۱۸	۱۷	۱۶	۱۵	۱۴	۱۳	۱۲	۱۱	۱۰	۹	۸	۷	۶	۵	۴	۳	۲	۱	۰
Type								**Code**								**Checksum**															
Parameters																															
Data																															

- فیلد **Type** : در این فیلد عددی قرار می‌گیرد که بیانگر نوع پیام می‌باشد و ساختار فیلدهای Parameters و Data بسته به عددی که در این فیلد قرار می‌گیرد متفاوت خواهد بود.
- فیلد **Code** : گاهی خود نوع پیام به چند زیر نوع دیگر تقسیم می‌شود که کد زیر نوع در این فیلد قرار می‌گیرد.

➢ فیلد **Checksum** : محتوای این فیلد برای سنجش اعتبار و سلامت بستهٔ ICMP مورد استفاده قرار می‌گیرد. تمامی بستهٔ ICMP بصورت دوبایت دوبایت جمع شده و نهایتاً از مکمل ۱ حاصل جمع ، عددی ۱۶ بیتی بدست می‌آید که درون این فیلد قرار می‌گیرد.

در ادامه نوع و ساختار پیامهای ICMP را توضیح می‌دهیم:

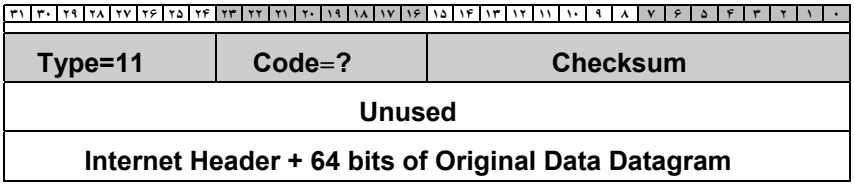
♦ پیام Destination Unreachable : این پیام زمانی صادر می‌شود که زیر شبکه یا یک مسیریاب نتواند آدرس مقصد را تشخیص بدهد یا به هر دلیلی بسته توسط ماشین میزبان تحویل گرفته نشود . (مثلاً بدلیل بزرگ بودن اندازهٔ بسته‌ها و عدم اجازه به مسیریاب برای شکستن آن)
ساختار بستهٔ حامل این پیام به صورت زیر است:



معنای شماره‌های مختلف در فیلد Code به شرح زیر است:
0 : شبکهٔ مورد نظر در دسترس نمی‌باشد.
1 : ماشین میزبان مورد نظر در دسترس نمی‌باشد.
2 : پروتکل مورد نظر تعریف نشده است.
3 : شمارهٔ پورت مورد نظر وجود ندارد.
4 : اندازهٔ بسته بزرگ است و نیاز به شکستن دارد در حالی که اجازه داده نشده است.

♦ پیام Time Exceeded : این پیام زمانی صادر می‌شود که مهلت قانونی یک بسته منقضی شده باشد و یک مسیریاب مجبور شود آنرا حذف کند؛ در چنین حالتی این پیام به آدرس فرستنده بسته IP برای آگاهی ارسال خواهد شد .

ساختار بستهٔ حامل این پیام به صورت زیر است:



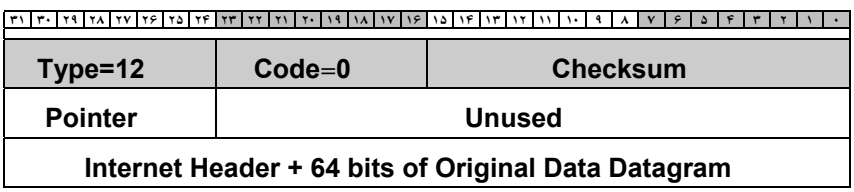
معنای شماره‌های مختلف در فیلد Code به شرح زیر است:

0 : زمان حیات بسته منقضی شده است. (این پیام معمولاً توسط مسیریاب صادر میشود)

1 : زمان بازسازی قطعات یک دیتاگرام منقضی شده است. (این پیام توسط ماشین میزبان صادر میشود)

♦ پیام Parameter Problem : این پیام زمانی صادر خواهد شد که مقداری نامعتبر در یکی از فیلدهای سرآیند در بسته IP قرار گرفته باشد و مسیریاب قادر به تشخیص و تفسیر سرآیند آن بسته IP نباشد. بعنوان مثال در فیلد Version از بسته IP عدد ۵ قرار گرفته باشد و یا Checksum با سرآیند تناقض داشته باشد.

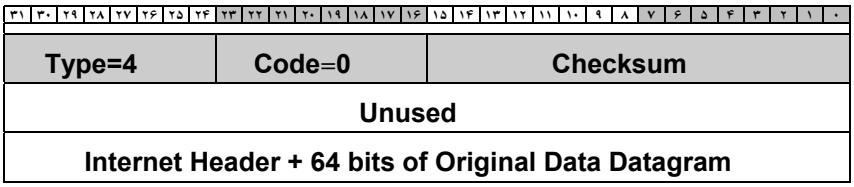
ساختار بستهٔ حامل این پیام به صورت زیر است:



فیلد Pointer محل بایتی را در بسته مشخص می‌کند که خطا در آن ناحیه بوده است.

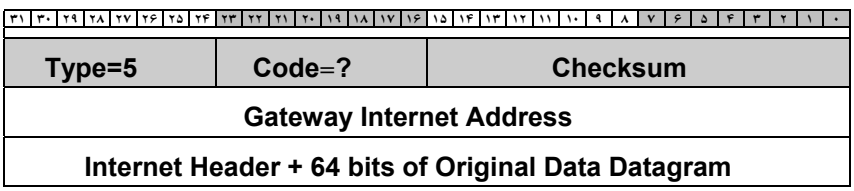
♦ پیام Source Quench : این بسته زمانی برای یک ماشین میزبان ارسال می‌شود که از آن خواسته شود حجم ارسال بسته‌هایش را کاهش بدهد چرا که در غیر اینصورت ازدحام پیش خواهد آمد . در مجموع هر گاه ازیک ماشین میزبان تقاضای کاهش نرخ تولید و ارسال

بسته‌های IP را داشته باشد این پیام را صادر می‌نماید. اگر ماشین میزبان پس از طی مدّت مشخصی این پیام را دریافت نکرد می‌تواند سرعت تولید بسته‌ها را به حالت اول برگرداند.
ساختار بستهٔ حامل این پیام به صورت زیر است:



♦ پیام Redirect : این پیام زمانی صادر می‌شود که یک مسیریاب احساس کند بسته یا بسته‌هایی که برای او ارسال شده است در مسیر صحیح نیستند و احتمالاً اشکالی در مسیریابی وجود دارد . این پیام می‌تواند برای هشدار خطاهای احتمالی موثر باشد.

ساختار بستهٔ حامل این پیام به صورت زیر است:



معنای شماره‌های مختلف در فیلد Code به شرح زیر است:

0 : باید تغییر مسیر به شبکه ای که آدرس آن مشخص شده است انجام شود.

1 : باید تغییر مسیر به ماشینی که آدرس آن مشخص شده است انجام شود.

2 : برای برآورده شدن سرویس ویژهٔ درخواستی که در فیلد Type of service مشخص شده ، باید تغییر مسیر به شبکه ای که آدرس آن مشخص شده است انجام شود.

3 : برای برآورده شدن سرویس ویژهٔ درخواستی که در فیلد Type of service مشخص شده ، باید تغییر مسیر به ماشینی که آدرس آن مشخص شده است انجام شود.

فرض کنید به مسیریاب R1 بسته ای ارسال شده و او با بررسی جدول مسیریابی آنرا به مسیریاب R2 فرستاده تا او آنرا به مقصد X برساند .حال اگر R2 با مقایسهٔ الگوی زیرشبکه به این نتیجه رسید که خود او و فرستنده آن بسته در یک شبکه واقعند با ارسال این پیام به فرستنده اعلام میکند اگر از این به بعد بسته‌هایش به جای اینکه به R1 ارسال شود به R2 داده شود ، زودتر به مقصد خواهد رسید؛ ضمناً آدرس IP خودش را نیز در فیلد Gateway Internet Address قرار می‌دهد .

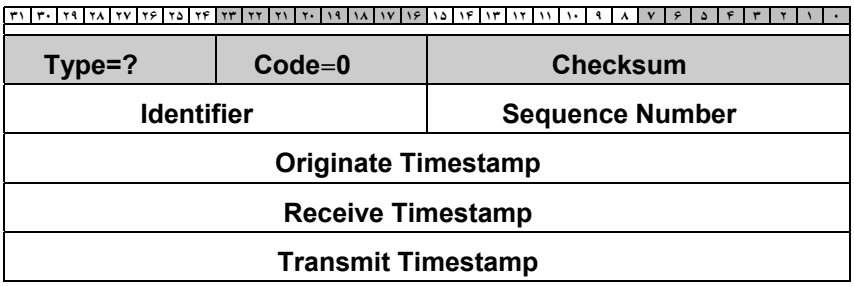
♦ پیغاهای Echo Request , Echo Reply : پیام Echo Request وقتی صادر می‌شود که یک مسیریاب بخواهد بداند آیا یک ماشین خاصّ شبکه قابل دسترس و موجود است یا خیر. در پاسخ به دریافت Echo Request ، مقصد با ارسال پیام Echo Reply به آن پاسخ می‌دهد . با این پرسش و پاسخ ، یک ماشین می‌تواند از قابل دسترس بودن یک مسیریاب یا ماشین میزبان در شبکه مطلع شود.
ساختار بستهٔ حامل این پیامها به صورت زیر است:

۳۱	۳۰	۲۹	۲۸	۲۷	۲۶	۲۵	۲۴	۲۳	۲۲	۲۱	۲۰	۱۹	۱۸	۱۷	۱۶	۱۵	۱۴	۱۳	۱۲	۱۱	۱۰	۹	۸	۷	۶	۵	۴	۳	۲	۱	۰
Type=?								Code=0								Checksum															
Identifier																Sequence Number															
Data																															

معنای شماره‌های مختلف در فیلد Type به شرح زیر است:
8 : برای مشخص کردن پیام Echo Request
0 : برای مشخص کردن پیام Echo Reply

ابتدا پیام Echo Request به سمت ماشین مقصد ارسال میشود و ماشینی که آنرا دریافت کند ، آدرسهای مبدا و مقصد را عوض کرده و شمارهٔ نوع آنرا از ۸ به صفر تغییر داده ، پس از محاسبهٔ مجدد کد کشف خطا ، آنرا برمیگرداند. فیلدهای Identifier و Sequence number برای پیشگیری از اشتباه در همخوانی و تطابق پیامهای رفت و برگشتی است تا مبدأ بداند یک پاسخ مربوط به کدام تقاضای اوست .

♦ پیامهای Timestamp Reply و Timestamp Request : این دو پیام دقیقاً شبیه دو پیام تعریف شده در قبل هستند با این تفاوت که دریافت کنندهٔ آن ، زمان دریافت و زمان ارسال بسته را نیز در پاسخ به آن اضافه خواهد کرد. بنابراین ارسال کننده پیام Timestamp Request پس از دریافت پاسخ نه تنها از قابل دسترس بودن مقصد باخبر می‌شود بلکه زمان رفت و برگشت یک بسته را نیز می‌تواند تخمین بزند و به کمک آن جداول مسیریابی و همچنین کارائی شبکه را اندازه گیری نماید.
ساختار بستهٔ حامل این پیامها به صورت زیر است:



معنای شماره‌های مختلف در فیلد Type به شرح زیر است:
13 : برای مشخص کردن پیام Timestamp Request
14 : برای مشخص کردن پیام Timestamp Reply

Identifier & Sequence Number همانند پیامهای قبلی برای پیشگیری از اشتباه در همخوانی و تطابق پیامهای رفت و برگشتی است. Originate Timestamp زمانی است که مبدأ ، آن پیام را ارسال کرده است (زمان بر حسب میلی ثانیه گذشته از نیمه شب و بر اساس زمان جهانی گرینویچ است) . Receive Timestamp زمانی است که گیرنده آن را دریافت کرده است و Transmit Timestamp زمان ارسال پاسخ بسته از طرف مقابل است . اگر زمان بر حسب میلی ثانیه آماده نبود بیت پرارزش از فیلد زمان یک می‌شود تا معلوم شود که آن فیلد معتبر نیست.

در پروتکل ICMP چهار پیام دیگر نیز وجود دارد که با استفاده از آنها یک ماشین میزبان می‌تواند آدرس IP شبکه محلی خود را در هنگامیکه چندین شبکه محلی از آدرسهای IP مشترک استفاده می‌کند پیدا نماید.

برای بدست آوردن اطلاعات جزئی تر و دقیق در مورد وظایف و پیامهای پروتکل ICMP به RFC-792 مراجعه نمائید.

۶) پروتکل ARP[1]

نکته ظریفی که در مورد شبکه اینترنت وجود دارد آن است که اگر چه تمامی ماشینهای میزبان و ابزارهای شبکه ای از آدرس IP که آدرس منحصر به فرد و یکتا است استفاده می‌کنند ولیکن یک بسته IP فقط در لایه شبکه قابل شناسائی و تحلیل است. یک بسته IP قبل از ارسال روی کانال از لایه اول یعنی لایهٔ فیزیکی عبور می‌کند و ضمن اضافه شدن اطلاعات لازم و تشکیل یک فریم ، روی کانال فیزیکی ارسال می‌شود . بعبارت روشنتر بسته IP قبل از ارسال درون فیلد داده از فریمی قرار می‌گیرد که بعداً در لایه اول تشکیل می‌شود؛ لایه اول وظیفه ای در قبال مسیریابی و کارهائی از این قبیل ندارد و فقط با آدرسهای فیزیکی کار می‌کند . بعنوان مثال اگر ماشین شما بخواهد بسته ای را برای ماشینی که روی شبکه محلی خودتان واقع است بفرستد، در لایه اول الزاماً بایستی آدرس فیزیکی ماشین شما (مبداء) و آدرس فیزیکی ماشین طرف مقابل (مقصد) معین باشد . (این آدرسها بصورت سخت افزاری در کارت شبکه درج شده است) عدم دانستن آدرسهای فیزیکی عملاً مساوی عدم توانایی برای ارتباط خواهد بود چرا که روی کانال انتقال آدرسهای IP بی معنا هستند .

حال فرض کنید ماشین شما می‌خواهد بسته ای را برای ماشین دیگر ارسال کند که روی شبکه فعلی شما نیست. در این حالت هم لایه اول یک فریم برای ارسال روی کانال فیزیکی تشکیل می‌دهد و نیاز به آدرس MAC از مقصد دارد؛ آدرس فیزیکی مقصد چیست؟

در لایه اول هر گاه بسته ای قرار است به خارج از شبکه ارسال شود آدرس فیزیکی مقصد ، آدرس مسیریاب پیش فرض شما خواهد بود . بنابراین آدرسهای MAC مقوله ای جدا هستند و آدرسهای IP مقوله ای دیگر.

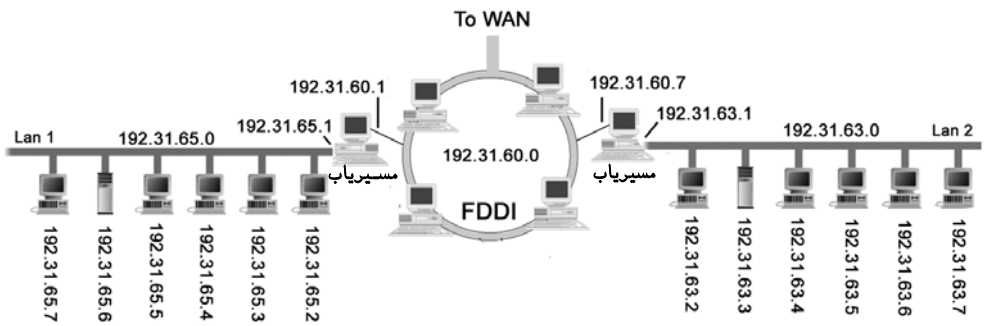
با مقدمه فوق به این نتیجه خواهیم رسید که هر ماشینی روی اینترنت گذشته از آن که بایستی آدرسهای IP خودش و مقصدش را بشناسد و بداند ، نیازمند به دانستن

[1] Address Resolution Protocol

آدرسهای فیزیکی ماشینهائی که مستقیماً با او در ارتباطند ، می‌باشد . بعنوان مثال شبکه اترنت که در تمام دنیا شناخته شده است از آدرسهائی استفاده می‌کند که منحصر به فرد و ٤٨ بیتی (٦ بایتی) است . بنابراین کامپیوتری که به یک کارت اترنت مجهز است گذشته از آن که بایستی یک آدرس IP منحصر به فرد داشته باشد یقیناً دارای یک آدرس ٤٨ بیتی یکتاست که این آدرس یکتا در کارخانه سازندهٔ آن ، تنظیم شده است . بنابراین وقتی پروتکل IP می‌خواهد یک بسته اطلاعاتی را روی شبکه بفرستد باید به نحوی آدرس فیزیکی اولین ماشینی که با آن بایستی ارتباط برقرار کند را بداند؛ این ماشین می‌تواند مسیریاب پیش فرض او باشد یا می‌تواند آدرس فیزیکی مقصد روی همین شبکه محلی باشد .

حال فرض کنید ایستگاهی آدرس IP ماشینی را که میخواهد با آن ارتباط برقرار کند ، می‌داند ولی آدرس فیزیکی او را نمی‌داند . چه کاری می‌تواند انجام بدهد؟ باید از پروتکل ARP بهره ببرد! در این پروتکل فرض بر آن است که تمامی ماشینهایِ روی یک شبکه محلی آدرس IP خود را می‌دانند.

برای روشن شدن وظیفهٔ پروتکل ARP به شکل (۱۰-۳) نگاه کنید.



شکل (۱۰-۳) شبکه بندی و آدرس دهی آنها در یک دانشکده

در مثال شکل (۱۰-۳) فرض کنید سه شبکه در دانشگاه شما نصب شده است . شبکهٔ محلی اول در دانشکدهٔ کامپیوتر با آدرس کلاس C به شماره 192.31.65.0 و شبکهٔ دوم در دانشکده برق با آدرس کلاس C به شمارهٔ 192.31.63.0 نصب شده است. (هر دو شبکه از نوع اترنت هستند)

این دو شبکه از طریق یک شبکه فیبر نوری با استاندارد FDDI و با آدرس IP شماره 192.31.60.0 به همدیگر متصل شده‌اند . هر ماشین در شبکهٔ اترنت یک آدرس

۴۸ بیتی یکتا دارد. مسیریابها در شکل مشخص شده‌اند و ارتباط دو شبکه اترنت را با FDDI برقرار می‌کنند. شبکهٔ FDDI از طریق یک خط اختصاصی به شبکهٔ جهانی اینترنت متصل شده است . هر مسیریاب به دو شبکهٔ متفاوت متصل شده و به عنوان عضوی از هر دو شبکه دارای دو آدرس IP مجزا می‌باشد، که هر یک از آنها در یکی از شبکه‌های محلی تعریف شده است .

حال فرض کنید که ماشینی مایل است به آدرس خاصی مثلاً 192.31.65.5 بستهٔ IP بفرستد. در لایهٔ شبکه یک بستهٔ IP با مشخصات لازم ساخته می‌شود و در قسمت آدرس مقصد مقدار 192.31.65.5 قرار می‌گیرد. از دیدگاه لایه شبکه پس از تشکیل بستهٔ IP ، کار تمام است و لیکن از دیدگاه لایه اول که بایستی آن بسته را روی کانال ارسال کند دانستن آدرس فیزیکی (آدرس MAC) ماشین مقصدی که آدرس IP آن 192.31.65.5 است ، حیاتی است .

وظیفه پروتکل ARP در اینجا آن است که یک "بسته فراگیر"[1] روی کل شبکه محلی منتشر کند که این بسته در حقیقت سوال می‌کند:

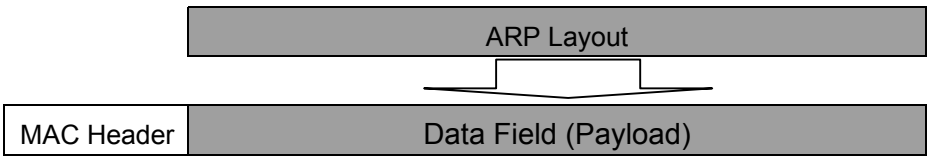
" آیکی IP او 192.31.65.5 است ، آدرس فیزیکی او چیست؟"

با توجه به آنکه بسته‌های فراگیر توسط تمامی ماشینهای روی شبکهٔ محلی دریافت می‌شود ، ماشینی که آدرس IP خودش را درون این بسته می‌بیند ، بدان پاسخ می‌دهد و آدرس فیزیکی خود را برای ارسال کنندهٔ آن بسته می‌فرستد. پس از آنکه آدرس فیزیکیِ مقصد بدست آمد ، یک فریم اترنت ساخته شده بر روی کانال منتقل می‌شود.

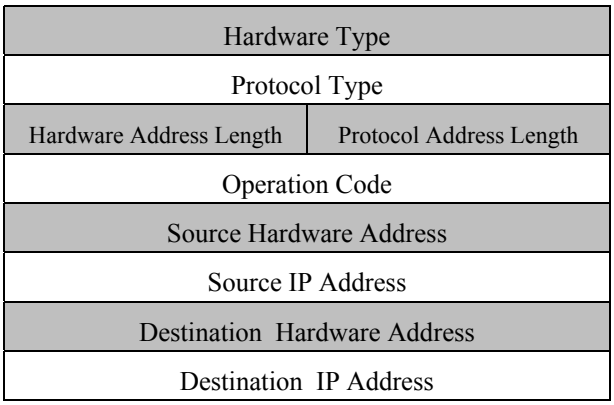
به این نکته توجه داشته باشید که هر ماشین بر روی شبکهٔ محلی از پروتکل ARP حمایت می‌کند و این پروتکل عملیات پرسش و پاسخ را برای هر ماشین که تقاضای ارسال بستهٔ IP دارد ، انجام می‌دهد.

بر خلاف پروتکل ICMP که روی پروتکل IP قرار می‌گیرد ، پروتکل ARP مستقیماً بر روی پروتکل لایهٔ فیزیکی عمل میکند؛ یعنی یک بستهٔ ARP ساخته شده و درون فیلد داده از فریم لایهٔ فیزیکی قرار گرفته و روی کانال ارسال می‌شود. در شکل (۱۱-۳) چگونگی ساخته شدن یک پیام ARP به تصویر کشیده شده است. در شکل (۱۲-۳) ساختار درونی بستهٔ ARP تشریح شده است.

[1] Broadcast



شکل (۱۱-۳) چگونگی قرار گرفتن یک پیام ARP درون فریم لایهٔ فیزیکی



شکل (۱۲-۳) ساختار پیامهای ARP

شماره نوع	عنوان سخت افزار کارت شبکه
1	**Ethernet**
2	**Experimental Ethernet**
3	**X.25**
4	**Proteon ProNET (Token Ring)**
5	**Chaos**
6	**IEEE 802.X**
7	**ARCnet**

جدول (۱۳-۳) تعریف استاندارد سخت افزار کارت شبکه

➢ **Hardware Type** : شمارهٔ نوع سخت افزار کارت شبکه که در لایهٔ اول وظیفهٔ انتقال اطلاعات روی کانال فیزیکی را بر عهده دارد. این شمارهها در جدول (۱۳-۳) مشخص شدهاند.

➢ **Protocol Type** : نوع پروتکلی که لایهٔ دوم از آن استفاده می‌شود. این پروتکلها و شمارهٔ آنها در جدول (۱٤-۳) مشخص شده‌اند. برای شبکه‌های مبتنی بر TCP/IP این شماره ۲۰٤۸ است.

➢ **Hardware Address Length** : با توجه به آنکه طول آدرسهای فیزیکی در شبکه‌ها ، متفاوت است در این فیلد طول آدرس (بر حسب بایت) مشخص میشود.

➢ **Protocol Address Length** : طول آدرسهای IP که در پروتکل TCP/IP مقدار ٤ است.

➢ **Operation Code (Opcode)** : ۱ برای ARP request

۲ برای ARP reply

➢ **Source Hardware Address** : آدرس فیزیکی مبدأ

➢ **Source IP Address** : آدرس IP ماشین مبدأ

➢ **Destination Hardware Address** : آدرس فیزیکی ماشین مقصد

➢ **Destination IP Address** : آدرس IP ماشین مقصد

برای بالا بردن سرعت پروتکل ARP ، وقتی برای یکبار آدرس فیزیکی متناظر با آدرس IP از یک ایستگاه بدست آمد ، پروتکل ARP این دو آدرس را در جدولی درون حافظهٔ اصلی که ARP Cache نامیده می‌شود ذخیره می‌کند تا اگر مجدداً به این آدرس نیاز شد به سرعت در اختیار قرار بگیرد . ساختار هر رکورد از این جدول بصورت زیر است:

IF Index	Physical Address	IP Address	Type

♦ **IF Index** : شماره پورت سخت افزاری متناظر با آن کارت شبکه

♦ **Physical Address** : آدرس سخت افزاری کارت شبکه

♦ **IP Address** : آدرس IP متناظر با آدرس سخت افزاری

♦ **Type** : مقداری که در این فیلد قرار می‌گیرد وضعیت هر رکورد را در این جدول مشخص میکند: مقدار ۱ : یعنی این رکورد باید بطور متناوب به هنگام شود. دقت کنید که ARP Cache هر دقیقه یکبار "بهنگام سازی"[۱] می‌شود. مقدار ٤: بدین معناست که این رکورد ثابت و بدون تغییر است و نباید بهنگام شود. مقدار ۱: یعنی رکورد چون بهنگام نشده از اعتبار ساقط است.

[۱] Update

شمارهٔ پروتکل	نام پروتکل
512	XEROX PUP
513	PUP Address Translation
1536	XEROX NS IDP
2048	Internet Protocol (IP)
2049	X.75
2050	NBS
2051	ECMA
2052	Chaosnet
2053	X.25 Level 3
2054	Address Resolution Protocol (ARP)
2055	XNS
4096	Berkeley Trailer
21000	BBN Simnet
24577	DEC MOP Dump/Load
24578	DEC MOP Remote Console
24579	DEC DECnet Phase IV
24580	DEC LAT
24582	DEC
24583	DEC
32773	HP Probe
32784	Excelan
32821	Reverse ARP
32824	DEC LANBridge
32823	AppleTalk

جدول (۱٤–۳) شمارهٔ پروتکلهای لایهٔ دوم

مسئله دیگری که ممکن است در هنگام بکارگیری پروتکل ARP رخ بدهد آن است که وقتی آدرس IP مربوط به ایستگاهی روی شبکه محلی سوال می‌شود ، ممکن است آن ایستگاه روی شبکه محلی دیگری باشد و بالطبع پاسخی نمی‌رسد. در چنین حالتی دو راه حل وجود دارد:

الف : وقتی مسیریابی که به آن شبکه متصل است می‌بیند آدرس مقصدی که توسط ARP سوال شده روی یک شبکه محلی دیگر واقع است در پاسخ به آن ، آدرس فیزیکی خودش را به ایستگاه سوال کننده ارسال می‌دارد؛ به این روش Proxy ARP گفته می‌شود.

ب: ایستگاهها خودشان موظفند به روشی که در مبحث "الگوی زیرشبکه" اشاره شد مستقلاً محلی بودن یا خارجی بودن ماشین مقصد را تشخیص داده و در صورت خارجی بودن ، آدرس فیزیکی یک مسیریاب مناسب را انتخاب نمایند.

نکته آخری که در مورد پروتکل ARP بایستی توضیح بدهیم آن است که در مسیریابها نیز برای شناسائی آدرس ایستگاههای یک شبکهٔ محلی متصل به آنها بهمین روش عمل می‌شود. برای جزئیات دقیقتر پروتکل ARP به REC-826 مراجعه کنید.

۷) پروتکل RARP[1]

پروتکل ARP برای یافتن آدرس‌های فیزیکی ایستگاههائی است که آدرس IP خود را می‌دانند. پروتکل RARP دقیقاً عکس پروتکل ARP عمل می‌کند . گاهاً اتفاق می‌افتد که ایستگاه آدرس فیزیکی مورد نظرش را میداند ولیکن آدرس IP آنرا نمی‌داند؛ این قضیه برای ایستگاههائی که بدون دیسکند و از طریق سرویس دهنده بوت می‌شوند صادق است .

در این پروتکل برای شناسایی آدرس IP متناظر با یک آدرس فیزیکی یک بسته فراگیر روی خط ارسال می‌شود که در آن آدرس فیزیکی یک ایستگاه قرار دارد . تمامی‌ایستگاههایی 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 بستهٔ RARP Reply برمی‌گردانند. بعنوان

[1] Reverse Address Resolution Protocol

مثال فرض کنید ایستگاهی با قرار دادن بستهٔ RARP و آدرس ٦ بایتی اترنت -14-04 D5-C8-01-25 روی خط ، آدرس IP آنرا طلب می‌کند . هر ماشین که آدرس IP متناظر با آن را می‌داند به این بسته RARP پاسخ می‌دهد.

دقت کنید که بسته‌های PARP,ARP از نوع "فراگیر محلی"[1] هستند و بالطبع توسط مسیریابها منتقل نمیشوند و فقط در محدوده شبکه محلی عمل می‌کنند. مستندات RARP در RFC903 آمده است .

۸) پروتکل BootP

با توجه به آنچه که در مورد RARP گفته شد بسته‌های سوال کننده آدرس IP از نوع محلی هستند و بالطبع این گونه بسته‌ها از مسیریابها به خارج از شبکه منتقل نخواهد شد.

گاهی نیاز است که یک آدرس IP روی چند شبکه محلی جستجو شود که در این حالت RARP جوابگو نیست. (این نیاز برای ایستگاههای بدون دیسک بوجود می‌آید چرا که پس از روشن شدن بایستی از طریق سرویس دهندهٔ شبکه[2] بوت شوند)

پروتکل BOOTP در چنین محیطهایی کاربرد دارد و از دیتاگرام‌های نوع UDP که در آینده به آنها خواهیم پرداخت استفاده می‌کند و مسیریابها موظف به انتقال آنها هستند . در این پروتکل نکته جالبی وجود دارد و آن هم آنست که در پاسخ به چنین بسته‌هائی به غیر از آدرس IP ایستگاه مورد نظر، اطلاعات لازم جهت بوت شدن سیستم و همچنین "الگوی زیرشبکه" برای ایستگاه تقاضا کننده که احتمالاً یک ایستگاه بدون دیسک است در قالب یک بستهٔ UDP ارسال خواهد شد.

۹) شمارهٔ پروتکل‌های استاندارد در لایهٔ سوم

دیتاگرامی که در فیلد داده از یک بستهٔ IP حمل می‌شود با ساختمان دادهٔ خاصّ از لایهٔ بالاتر تحویلِ پروتکل IP می‌شود تا روی شبکه ارسال شود. بعنوان مثال ممکن است این داده‌ها را پروتکل TCP در لایه بالاتر ارسال کرده باشد و یا ممکن است این

---

[1] Local Broadcast
[2] Network Server

کار توسط پروتکل UDP انجام شده باشد. بنابراین مقدار این فیلد شمارهٔ پروتکلی است که در لایه بالاتر تقاضای ارسال یک دیتاگرام کرده است؛ بسته‌ها پس از دریافت در مقصد باید به پروسهٔ متناظر با پروتکل تعیین شده ، تحویل داده شود. پروتکلهای لایهٔ بالاتر دارای یک شمارهٔ هشت بیتی منحصر بفرد و استاندارد هستند که در جدول (۱۵-۳) شماره و نام این پروتکلها ارائه شده است.

0	Reserved	[JBP]
1	ICMP	Internet Control Message [RFC792,JBP]
2	IGMP	Internet Group Management [RFC1112,JBP]
3	GGP	Gateway-to-Gateway [RFC823,MB]
4	IP	IP in IP (encasulation) [JBP]
5	ST	Stream [RFC1190,IEN119,JWF]
6	TCP	Transmission Control [RFC793,JBP]
7	UCL	UCL [PK]
8	EGP	Exterior Gateway Protocol [RFC888,DLM1]
9	IGP	any private interior gateway [JBP]
10	BBN-RCC-MON	BBN RCC Monitoring [SGC]
11	NVP-II	Network Voice Protocol [RFC741,SC3]
12	PUP	PUP [PUP,XEROX]
13	ARGUS	ARGUS [RWS4]
14	EMCON	EMCON [BN7]
15	XNET	Cross Net Debugger [IEN158,JFH2]
16	CHAOS	Chaos [NC3]
17	UDP	User Datagram [RFC768,JBP]
18	MUX	Multiplexing [IEN90,JBP]
19	DCN-MEAS	DCN Measurement Subsystems [DLM1]
20	HMP	Host Monitoring [RFC869,RH6]
21	PRM	Packet Radio Measurement [ZSU]
22	XNS-IDP	XEROX NS IDP [ETHERNET,XEROX]
23	TRUNK-1	Trunk-1 [BWB6]
24	TRUNK-2	Trunk-2 [BWB6]
25	LEAF-1	Leaf-1 [BWB6]
26	LEAF-2	Leaf-2 [BWB6]
27	RDP	Reliable Data Protocol [RFC908,RH6]
28	IRTP	Internet Reliable Transaction [RFC938,TXM]
29	ISO-TP4	ISO Transport Protocol Class 4 [RFC905,RC77]
30	NETBLT	Bulk Data Transfer Protocol [RFC969,DDC1]
31	MFE-NSP	MFE Network Services Protocol [MFENET,BCH2]
32	MERIT-INP	MERIT Internodal Protocol [HWB]
33	SEP	Sequential Exchange Protocol [JC120]
34	3PC	Third Party Connect Protocol [SAF3]
35	IDPR	Inter-Domain Policy Routing Protocol [MXS1]
36	XTP	XTP [GXC]
37	DDP	Datagram Delivery Protocol [WXC]
38	IDPR-CMTP	IDPR Control Message Transport Proto [MXS1]
39	TP++	TP++ Transport Protocol [DXF]
40	IL	IL Transport Protocol [DXP2]
41	SIP	Simple Internet Protocol [SXD]
42	SDRP	Source Demand Routing Protocol [DXE1]
43	SIP-SR	SIP Source Route [SXD]

44	SIP-FRAG	SIP Fragment	[SXD]
45	IDRP	Inter-Domain Routing Protocol	[Sue Hares]
46	RSVP	Reservation Protocol	[Bob Braden]
47	GRE	General Routing Encapsulation	[Tony Li]
48	MHRP	Mobile Host Routing Protocol	[David Johnson]
49	BNA	BNA	[Gary Salamon]
50	SIPP-ESP	SIPP Encap Security Payload	[Steve Deering]
51	SIPP-AH	SIPP Authentication Header	[Steve Deering]
52	I-NLSP	Integrated Net Layer Security TUBA	[GLENN]
53	SWIPE	IP with Encryption	[JI6]
54	NHRP	NBMA Next Hop Resolution Protocol	
55-60		Unassigned	[JBP]
61		any host internal protocol	[JBP]
62	CFTP	CFTP	[CFTP,HCF2]
63		any local network	[JBP]
64	SAT-EXPAK	SATNET and Backroom EXPAK	[SHB]
65	KRYPTOLAN	Kryptolan	[PXL1]
66	RVD	MIT Remote Virtual Disk Protocol	[MBG]
67	IPPC	Internet Pluribus Packet Core	[SHB]
68		any distributed file system	[JBP]
69	SAT-MON	SATNET Monitoring	[SHB]
70	VISA	VISA Protocol	[GXT1]
71	IPCV	Internet Packet Core Utility	[SHB]
72	CPNX	Computer Protocol Network Executive	[DXM2]
73	CPHB	Computer Protocol Heart Beat	[DXM2]
74	WSN	Wang Span Network	[VXD]
75	PVP	Packet Video Protocol	[SC3]
76	BR-SAT-MON	Backroom SATNET Monitoring	[SHB]
77	SUN-ND	SUN ND PROTOCOL-Temporary	[WM3]
78	WB-MON	WIDEBAND Monitoring	[SHB]
79	WB-EXPAK	WIDEBAND EXPAK	[SHB]
80	ISO-IP	ISO Internet Protocol	[MTR]
81	VMTP	VMTP	[DRC3]
82	SECURE-VMTP	SECURE-VMTP	[DRC3]
83	VINES	VINES	[BXH]
84	TTP	TTP	[JXS]
85	NSFNET-IGP	NSFNET-IGP	[HWB]
86	DGP	Dissimilar Gateway Protocol	[DGP,ML109]
87	TCF	TCF	[GAL5]
88	IGRP	IGRP	[CISCO,GXS]
89	OSPFIGP	OSPFIGP	[RFC1583,JTM4]
90	Sprite-RPC	Sprite RPC Protocol	[SPRITE,BXW]
91	LARP	Locus Address Resolution Protocol	[BXH]
92	MTP	Multicast Transport Protocol	[SXA]
93	AX.25	AX.25 Frames	[BK29]
94	IPIP	IP-within-IP Encapsulation Protocol	[JI6]
95	MICP	Mobile Internetworking Control Pro.	[JI6]
96	SCC-SP	Semaphore Communications Sec. Pro.	[HXH]
97	ETHERIP	Ethernet-within-IP Encapsulation	[RXH1]
98	ENCAP	Encapsulation Header	[RFC1241,RXB3]
99		any private encryption scheme	[JBP]
100	GMTP	GMTP	[RXB5]
101-254		Unassigned	[JBP]

جدول (۱۵–۳) شماره و نام پروتکلهای استانداردِ تولیدکننده و دریافت کنندهٔ دیتاگرام

۱۰) مراجع این فصل

مجموعهٔ مراجع زیر می‌توانند برای دست آوردن جزییات دقیق و تحقیق جامع در مورد مفاهیم معرفی شده در این فصل مفید واقع شوند.

RFC 1219	"On the Assignment of Subnet Numbers," Tsuchiya, P.F.; 1991
RFC 1112	"Host Extensions for IP Multicasting," Deering, S.E.; 1989
RFC 1088	"Standard for the Transmission of IP Datagrams over NetBIOS Networks," McLaughlin, L.J.; 1989
RFC 950	"Internet Standard Subnetting Procedure," Mogul, J.C.; Postel, J.B.; 1985
RFC 932	"Subnetwork Addressing Schema," Clark, D.D.; 1985
RFC 922	"Broadcasting Internet Datagrams in the Presence of Subnets," Mogul, J.C.; 1984
RFC 919	"Broadcasting Internet Datagrams," Mogul, J.C.; 1984
RFC 886	"Proposed Standard for Message Header Munging," Rose, M.T.; 1983
RFC 815	"IP Datagram Reassembly Algorithms," Clark, D.D.; 1982
RFC 814	"Names, Addresses, Ports, and Routes," Clark, D.D.; 1982
RFC 792	"Internet Control Message Protocol," Postel, J.B.; 1981
RFC 791	"Internet Protocol," Postel, J.B.; 1981
RFC 781	"Specification of the Internet Protocol (IP) Timestamp Option," Su, Z.; 1981