فصل دوم : مسیریابی در شبکه اینترنت

هدفهای آموزشی :

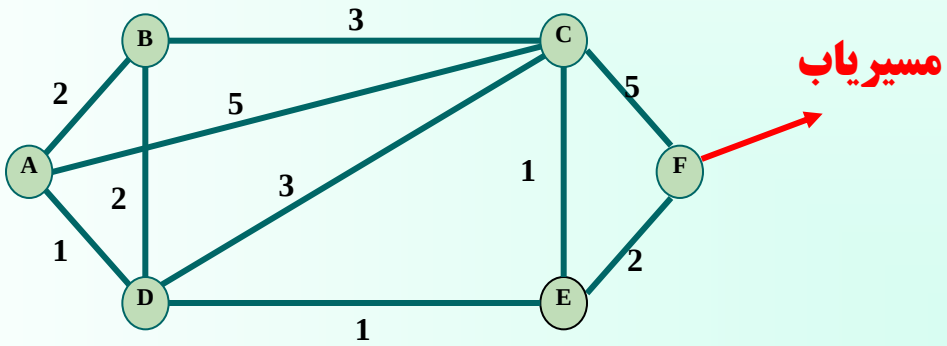
- مفاهیم اولیه مسیریابی
- الگوریتم‌های مسیریابی LS
- الگوریتم‌های مسیریابی بردار فاصله - DV -
- مسیریابی سلسله مراتبی
- پروتکل RIP
- پروتکل OSPF
- پروتکل BGP

۱) مفاهیم اولیه مسیریابی

مسیریاب: ابزاري است براي برقراري ارتباط دو يا چند شبكه

زيرساخت ارتباطي: مجموعه مسيريابها و كانالهاي فيزيكي ما بين آنها

الگوريتم‌هاي مسيريابي : روشهايي براي پيدا كردن مسيري بهينه ميان دومسيرياب به گونه‌اي كه هزينه كل مسير به حداقل برسد.



زیرساخت ارتباطی یک شبکهٔ فرضی

برخی اصطلاحات کلیدی در مسیریابی

آدرسهاي MAC:

- آدرسهاي لايه فيزيكي جهت انتقال فريمها بر روي كانال
- اندازه آدرس وابسته به پروتكل و توپولوژي شبكه
- تغيير آدرسهاي MAC بسته‌هاي اطلاعاتي هنگام عبور از مسيريابهاي موجود در مسير

آدرسهاي IP :

- آدرسهاي جهاني و منحصر به فرد
- مشخص‌كننده يک ماشين فارغ از نوع سخت افزار و نرم افزار آن
- ثابت بودن آدرسهاي IP بسته هاي اطلاعاتي هنگام عبور از مسيريابهاي موجود در مسير

بسته IP:

- واحد اطلاعاتي با اندازه محدود

توپولوژي شبكه:

- مجموعه مسيريابها و كانالهاي فيزيكي ما بين آنها در زيرساخت ارتباطي يك شبكه
- متغير با زمان

ترافيك شبكه:

- تعداد متوسط بسته‌هاي اطلاعاتي ارسالي و يا دريافتي روي يك كانال در واحد زمان
- متغير با زمان

گام يا Hop:

- عبور بسته از يك مسيرياب = گام
- تعداد مسيريابهاي موجود در مسير يك بسته = تعداد گام = **Hop Count**

ازدحام يا Congestion:

بيشتر بودن تعداد متوسط بسته‌هاي ورودي به يك مسيرياب از تعداد متوسط بسته هاي خروجي

بن بست Deadlock:

پايان طول عمر بسته‌ها

۱–۱) روشهای هدایت بسته‌های اطلاعاتی در شبکه‌های کامپیوتری

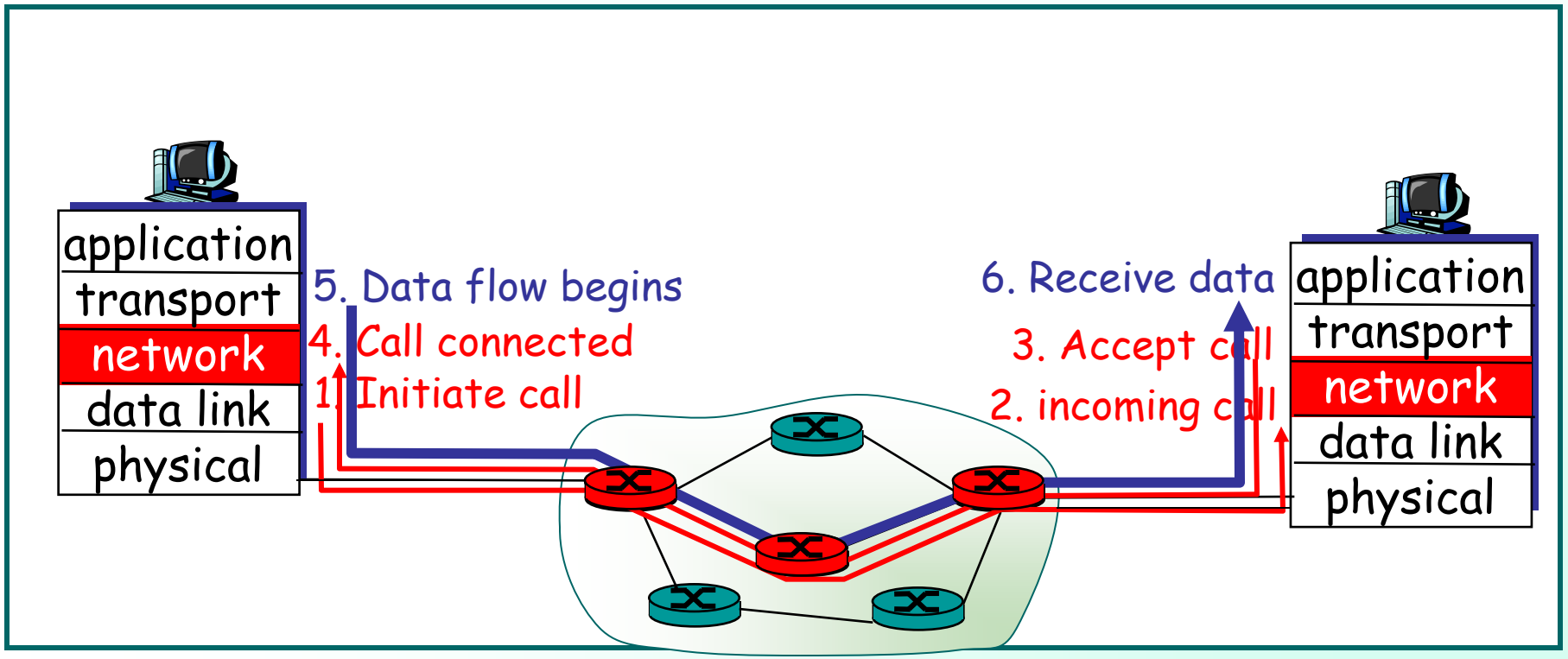
الف) روش مدار مجازي Virtual Circuit (VC)

ب) روش ديتاگرام Datagram

خصوصیات روش VC

- ارسال بسته‌هاي اطلاعاتي بدون نياز به اطلاع از آدرسهاي IP مبدأ و مقصد و فقط داشتن شماره VC جهت ارسال بسته
- عدم اجراي الگوريتم مسيريابي جهت هدايت بسته‌هاي اطلاعاتي از مبدأ به مقصد
- دريافت بسته به ترتيب ارسال شده در مقصد
- عدم احتمال گم‌شدن بسته‌ها در عمل مسيريابي در شبكه

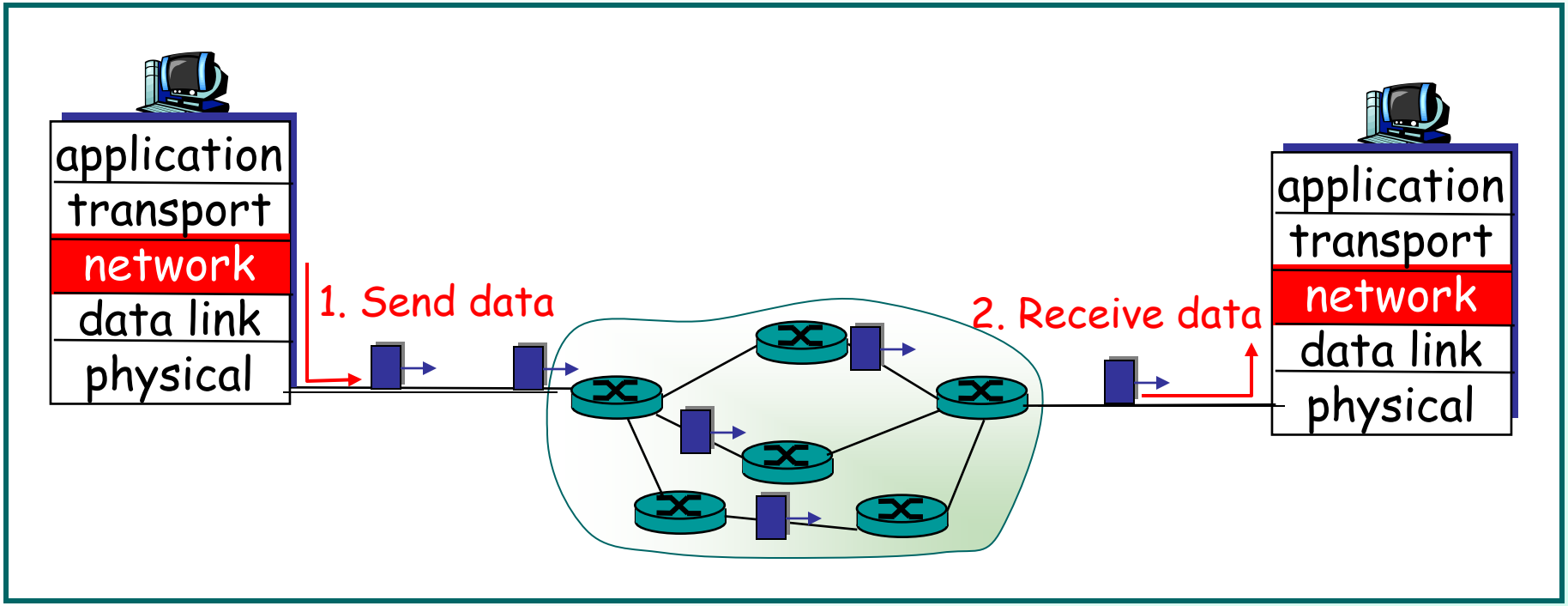
روش vc


application
transport
network
data link
physical
5. Data flow begins
4. Call connected
1. Initiate call
6. Receive data
3. Accept call
2. incoming call
application
transport
network
data link
physical

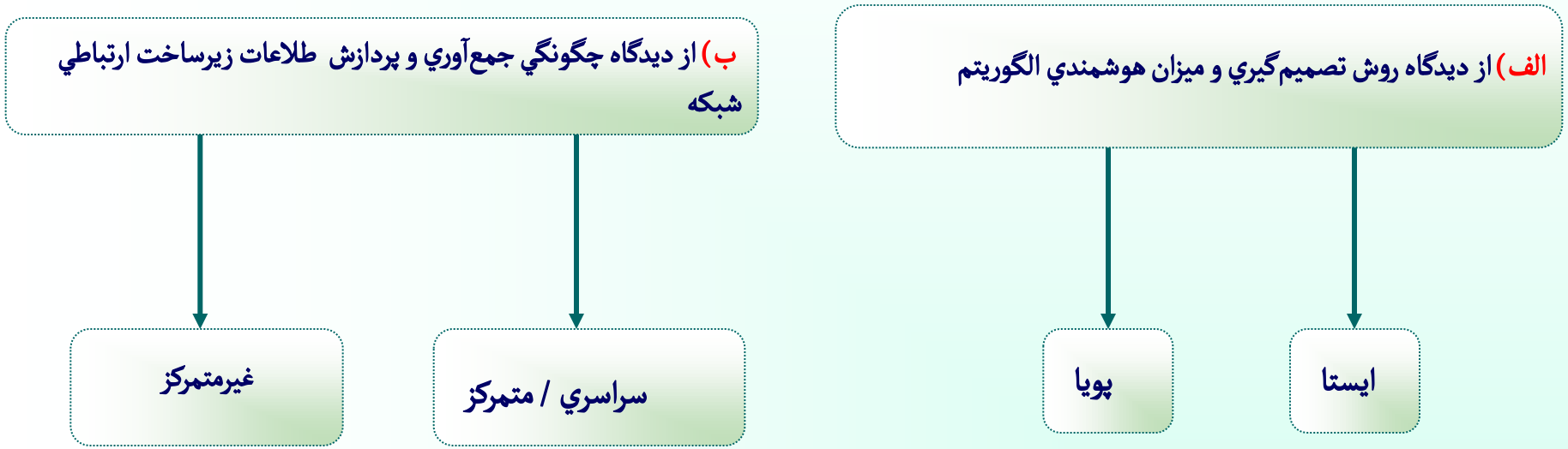
خصوصيات روش ديتاگرام

- ارسال بسته‌هاي اطلاعاتي با استفاده از آدرسهاي IP مبدأ و مقصد در شبكه
- انجام مسيريابي جداگانه براي هر بسته
- توزيع و هدايت بسته‌ها روي مسيرهاي متفاوت بر اساس شرايط توپولوژيكي و ترافيكي لحظه‌اي شبكه
- امكان دريافت بسته بدون ترتيب ارسال شده در مقصد
- لزوم نظارتهاي ويژه بر گم شدن و يا تكراري بودن بسته در لايه‌هاي بالاتر

روش Datagram



انواع الگوریتمهای مسیریابی



الگوريتم ايستا

- عدم توجه به شرايط توپولوژيكي و ترافيك لحظه‌اي شبكه
- جداول ثابت مسيريابي هر مسيرياب در طول زمان
- الگوريتم‌هاي سريع
- تنظيم جداول مسيريابي به طور دستي در صورت تغيير توپولوژي زيرساخت شبكه
- تغيير مسيرها به كندي در اثناي زمان

الگوريتم پويا

- به هنگام سازي جداول مسيريابي به صورت دوره‌اي بر اساس آخرين وضعيت توپولوژيكي و ترافيك شبكه
- تغيير سريع مسيرها
- تصميم‌گيري بر اساس وضعيت فعلي شبكه جهت انتخاب بهترين مسير
- ✕ ايجاد تأخيرهاي بحراني هنگام تصميم‌گيري بهترين مسير به جهت پيچيدگي الگوريتم

الگوريتم سراسری

- اطلاع كامل تمام مسيريابها از همبندي شبكه و هزينه هر خط
- الگوريتم‌هاي Link State (LS)

الگوريتم غير متمركز

- محاسبه و ارزيابي هزينه ارتباط با مسيريابهاي همسايه (مسيريابهايي كه به صورت مستقيم و فيزيكي با آن در ارتباط هستند)
- ارسال جداول مسيريابي توسط هر مسيرياب در فواصل زماني منظم براي مسيريابهاي مجاور
- پيچيدگي زماني كم
- الگوريتم‌هاي Distance Vector

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

- سريعترين الگوريتم براي ارسال اطلاعات به مقصد در شبكه
- جهت ارسال بسته‌هاي فراگير و كنترلي مانند اعلام جداول مسيريابي

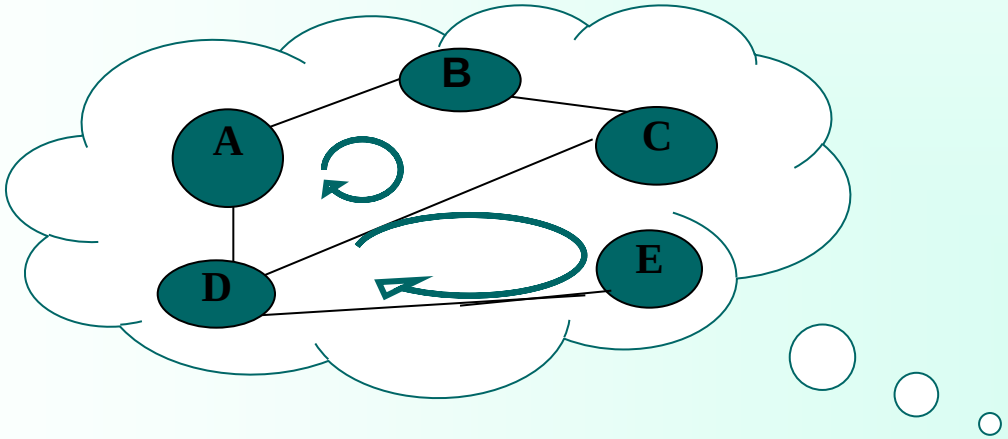
مشکل روش سیل آسا

- ایجاد حلقه بینهایت و از کارافتادن شبکه

راه حل رفع مشکل حلقه بینهایت

1) قراردادن شماره شناسايي براي هر بسته Selective Flooding

2) قراردادن طول عمر براي بسته‌ها



حلقه‌هاي بينهايت در روش سيل آسا

الگوريتم هاى LS

1- شناسايي مسيريابهاي مجاور

2- اندازه‌گيري هزينه

3- تشكيل بسته‌هاي LS

4- توزيع بسته‌هاي LS روي شبكه

5- محاسبه مسيرهاي جديد

1- شناسايي مسيريابهاي مجاور

- ارسال بسته خاصي به نام بسته سلام Hello Packet توسط مسيرياب به تمام خروجي‌ها
- پاسخگويي مسيريابهاي متصل از طريق كانال فيزيكي مستقيم به بسته ارسالي و اعلام آدرس IP خود به مسيرياب
- درج اطلاعات بسته‌هاي پاسخ در جدول مسيرياب

2- اندازه‌گيري هزينه

- اندازه‌گيري تأخير هر يك از خطوط خروجي مسيرياب توسط خود مسيرياب
- ارسال بسته خاص به نام Echo Packet روي تمام خطوط خروجي خود
- پاسخ تمام مسيريابهاي گيرنده بسته با ارسال بسته Echo Reply
- اگر مسيرياب موظف باشد كه با دريافت بستة Echo خارج از نوبت و به سرعت به آن پاسخ بدهد ، "زمان رفت و برگشت" اين بسته فقط تاخير فيزيكي بين دو مسيرياب را به عنوان معيار هزينه مشخص مي‌كند.
- اندازه‌گيري اين زمان با استفاده از زمان سنج و تقسيم آن مقدار بر عدد 2 و درج در جدول توسط مسيرياب

3- تشكيل بسته‌هاي LS

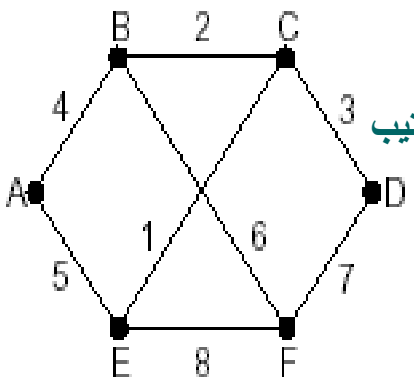
تشكيل بسته LS پس از جمع آوري اطلاعات لازم از مسيريابهاي مجاور شامل:

الف) آدرس جهاني مسيرياب توليدكنندة بسته

ب) يك شمارة ترتيب (تا بسته‌هاي تكراري از بسته‌هاي جديد تشخيص داده شوند.)

ج) طول عمر بسته (تا اطلاعات بسته ، زمان انقضاي اعتبار داشته باشد.)

د) آدرس جهاني مسيريابهاي مجاور و هزينة تخميني



فيلد شماره ترتيب

فيلد طول عمر

A	
Seq.	
Age	
B	4
E	5

Link	
B	
Seq.	
Age	
A	4
C	2
F	6

State	
C	
Seq.	
Age	
B	2
D	3
E	1

D	
Seq.	
Age	
C	3
F	7

Packets	
E	
Seq.	
Age	
A	5
C	1
F	8

F	
Seq.	
Age	
B	6
D	7
E	8

يك زيرساخت از يك شبكه فرضي

بسته‌هاي LS

4- توزيع بسته‌هاي LS روي شبكه

- ارسال بسته‌هاي LS به روش سيل آسا
- وجود شماره ترتيب براي هر بسته جهت جلوگيري از بروز حلقه تكرار
- در نظرگرفتن طول عمر براي هر بسته جهت رفع مشكل دريافت بسته‌هاي تكراري
- احراز هويت ارسال‌كننده بسته LS در مسيريابها جهت جلوگيري از بسته‌هاي LS آلوده

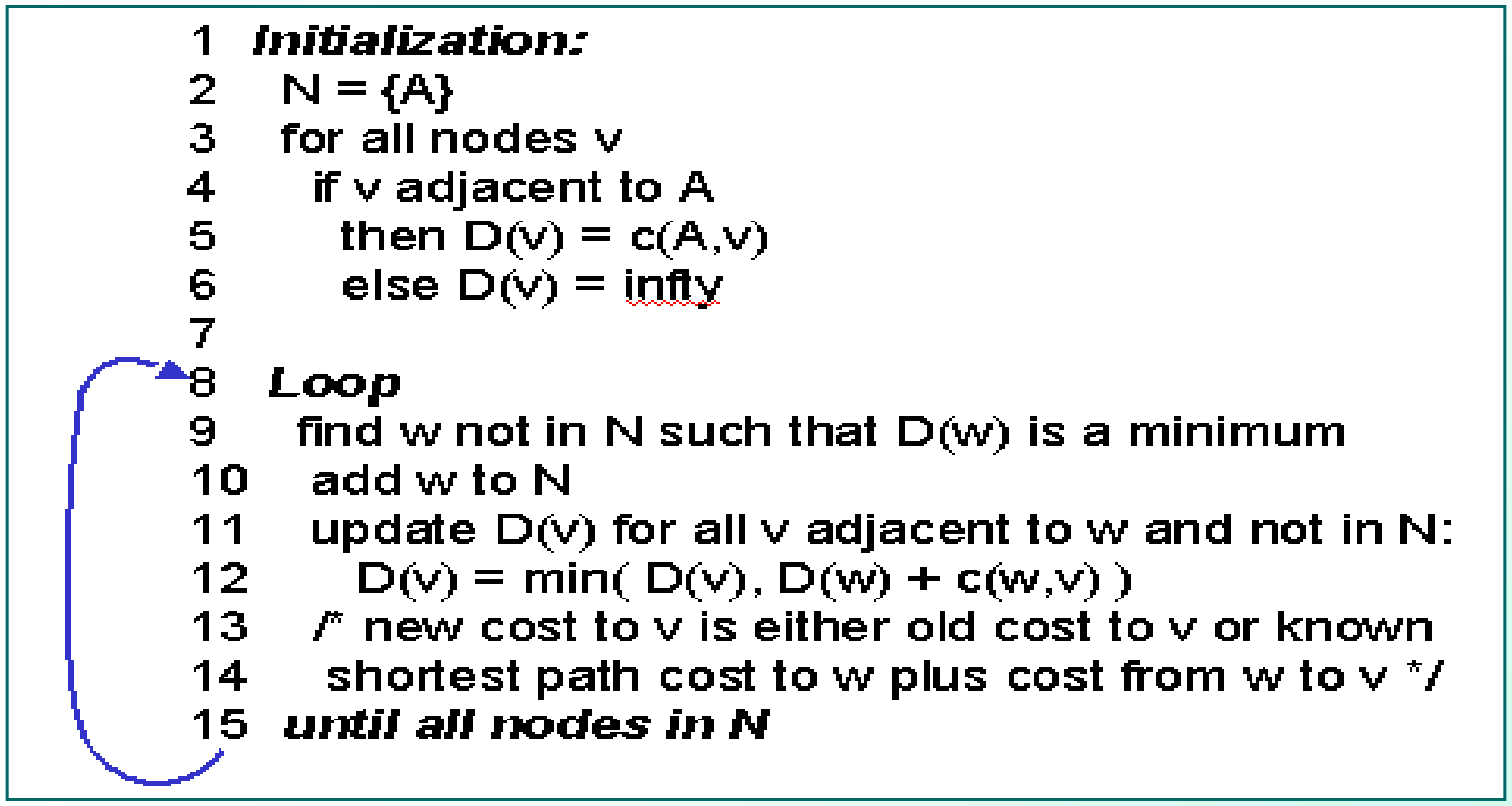
5- محاسبه مسيرهاي جديد

- تشكيل ساختمان داده گراف زيرشبكه جهت انتخاب بهترين مسير بين دو گره هنگام دريافت بسته‌هاي LS از تمام مسيريابهاي شبكه
- استفاده از الگوريتم دايجكسترا جهت يافتن بهترين مسير بين دو گره

(Dijkstra Shortest Path Algorithm)

* C(i , j) بيانگر هزينه خط ميان گره i تا j است.
  هرگاه همسايگاني در مجاورت گره وجود نداشته باشند
  C(i , j) بينهايت تلقي مي شود.
* D(v) هزينه فعلي مسير ميان مبدا تا گره V.
* P(v) گره‌اي كه در طول مسير از مبدا تا V درست قبل از V واقع شده.
* N مجموعه گره‌هايي كه عبور از آنها كم هزينه برآورد گشته است.

Dijkstra's Algorithm



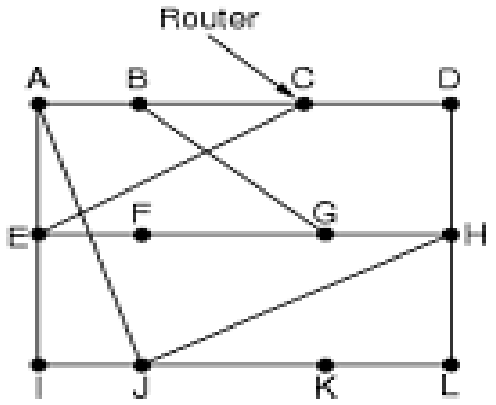
الگوریتمهای DV یا بردار فاصله

- يكي از روشاي پويا در مسيريابي
- مورد استفاده در شبكه ARPA
- استفاده در مسيريابهاي كوچك
- نامهاي متفاوت روش DV
- پروتكل RIP
- الگوريتم مسيريابي Bellman - Ford
- الگوريتم مسيريابي Ford – Fulkerson
- الگوريتم Distance Vector Routing

اصول کار روش DV

- محاسبه خطوطي را كه به صورت فيزيكي با مسيريابهاي ديگر دارد و درج در جدول مسيريابي
- بينهايت درنظرگرفتن هزينة خطوطي كه مسيرياب با آنها در ارتباط مستقيم نيست
- ارسال ستون هزينه از جدول مسيريابي براي مسيريابهاي مجاور در بازه‌هاي زماني مشخص ، توسط هر مسيرياب (“يعني فقط براي مسيريابهائي كه با آن در ارتباط است نه تمام مسيريابها ”). دريافت اطلاعات جديد ا زمسيريابهاي مجاور در در فواصل T ثانيه‌اي
- به هنگام نمودن جدول مسيريابي پس از دريافت جداول مسيريابي از مسيريابهاي مجاور ، طبق يك الگوريتم بسيار ساده

الگوریتمهای DV یا بردار فاصله



(a)

زیرساخت ارتباطی یک شبکهٔ فرضی
با دوازده مسیریاب

To	A	I	H	K	New estimated delay from J	Line
A	0	24	20	21	8	A
B	12	36	31	28	20	A
C	25	18	19	36	28	I
D	40	27	8	24	20	H
E	14	7	30	22	17	I
F	23	20	19	40	30	I
G	18	31	6	31	18	H
H	17	20	0	19	12	H
I	21	0	14	22	10	I
J	9	11	7	10	0	—
K	24	22	22	0	6	K
L	29	33	9	9	15	K
	JA delay is 8	JI delay is 10	JH delay is 12	JK delay is 6	New routing table for J	

Vectors received from J's four neighbors

(b)

جدول مسیریابی مربوط به مسیریاب J

مشکل عمده پروتکلهای DV

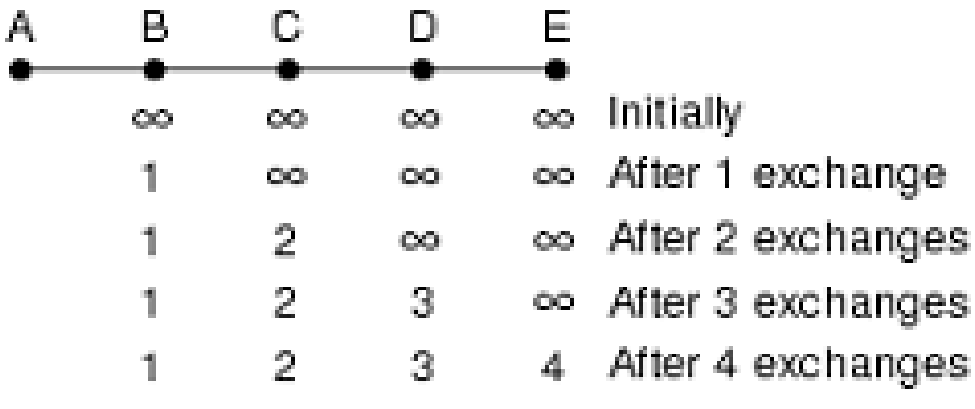
عدم همگرايي سريع جداول مسيريابي هنگام خرابي يك مسيرياب يا يك كانال ارتباطي = مشكل شمارش تا بينهايت

راه حل :

وقتي يك مسيرياب مي‌خواهد اطلاعاتي را به همسايه‌هايش بدهد هزينه رسيدن به آنهايي را كه قطعاً بايد از همان مسيرياب بگذرند را اعلام نمي‌كند. (يا ∞ اعلام مي‌كنند)

مسئله شمارش تا بينهايت

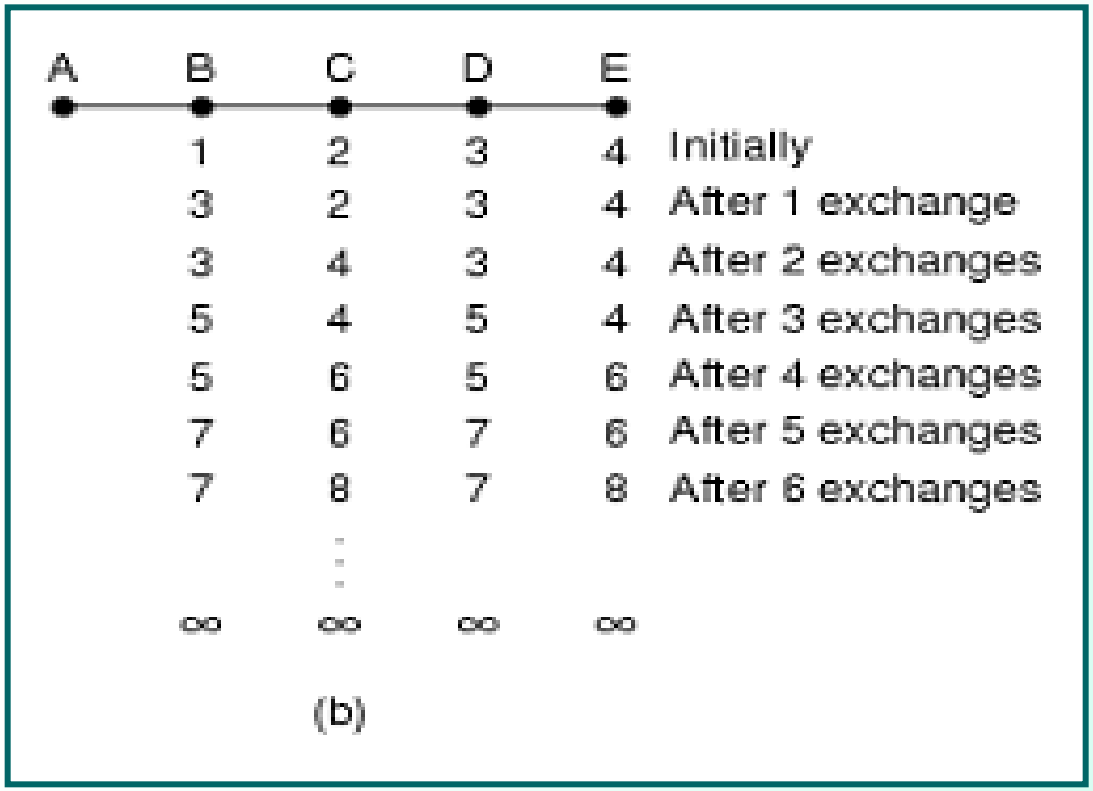
به خبرهاي خوب واكنش سريع ولي به خبرهاي بد واكنش كندي نشان مي دهد.



The count-to-infinity problem.

مسئله شمارش تا بینهایت

هرگاه مسيريابي از زيرشبكه خارج شود هركدام از ساير مسيرياب‌هاي فعال احساس مي‌كنند از طريق ديگري مسيري بهتر به آن وجود دارد.

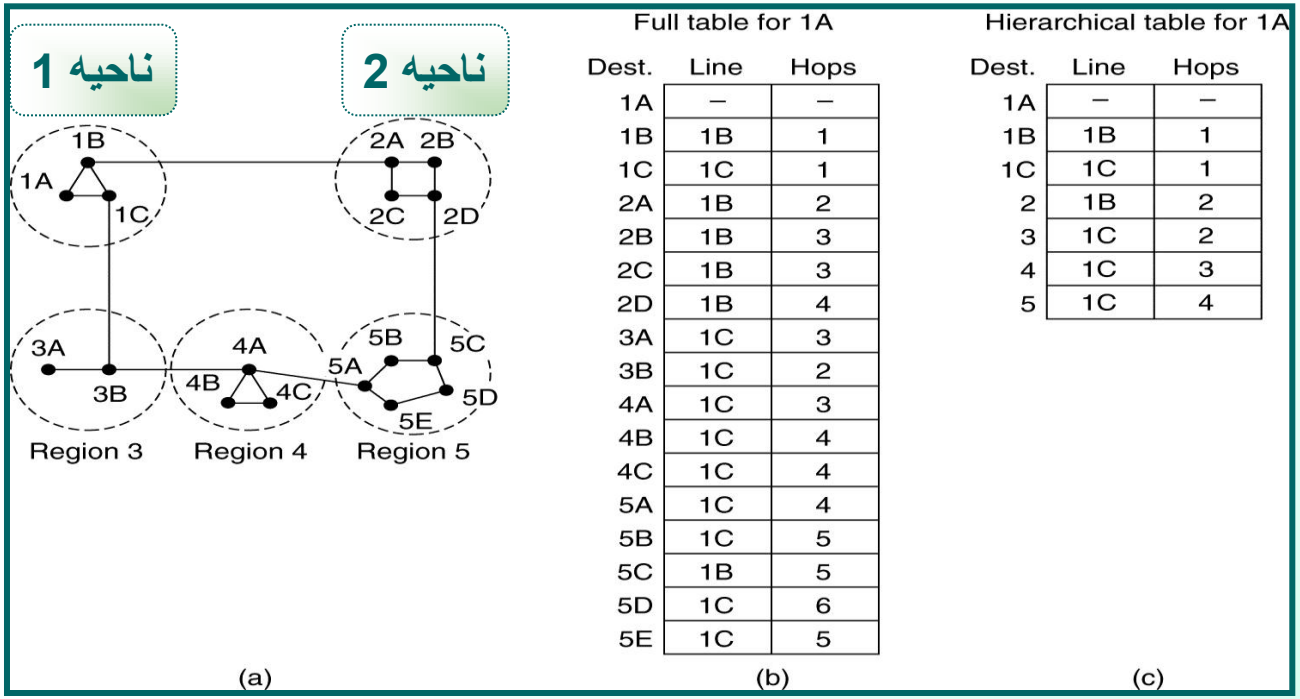


مسیریابی سلسله‌مراتبی Hierarchical Routing

رشد شبکه و زیادشدن شبکه‌هاي محلي و مسیریابها ، افزایش حجم جداول مسیریابي و زیادشدن زمان لازم جهت تعیین مسیر یك بسته و درنتیجه ایجاد تأخیرهاي بحراني و کاهش کارآیي شبکه

در مسیریابي سلسله‌مراتبي ، مسیریابها در گروههایي به نام "ناحیه Region" دسته‌بندي مي‌شوند. هر مسیریاب فقط "نواحي" و مسیریابهاي درون ناحیة خود را مي‌شناسد و هیچ اطلاعي از مسیریابهاي درون نواحي دیگر ندارد.

مسیریابی سلسله‌مراتبی



مشكل روش سلسله مراتبي

به دليل مشخص‌نبودن كل توپولوژي زيرشبكه براي هر مسيرياب :

ممكن است مسير انتخابي جهت ارسال بسته به يك مسيرياب خاص درون يك ناحيه بهينه نباشد.

مزيت استفاده از روشهای سلسله مراتبی: صرفه جويي در اندازه جداول مسيريابي

	تعداد ناحیه Regions	تعداد دسته Clusters	تعداد حوزه Zones	تعداد مسیریاب	تعداد رکورد در جدول
مسیریابی DV بدون سلسله‌مراتب	۱	–	–	۷۲۰	۷۲۰
مسیریابی DV با سلسله‌مراتب دوسطحی	۲۴	–	–	۳۰	۵۳
مسیریابی DV با سلسله‌مراتب سه‌سطحی	۹	۸	–	۱۰	۲۵
مسیریابی DV با سلسله‌مراتب سه‌سطحی	۹	۵	۴	۴	۱۹

مقایسه اندازه جدول مسیریابی در روشهای سلسله مراتبی

مسیریابی در اینترنت

اینترنت مجموعه‌اي از شبکه‌هاي خودمختار Autonomous و "مستقل" است که به نحوي به هم متصل شده‌اند.
شبکهٔ خودمختار که اختصاراً AS نامیده مي‌شود ، شبکه‌اي است که تحت نظارت و سرپرستي يك مجموعه يا سازمان خاص پياده و اداره مي‌شود. مثلاً يك دانشگاه

مسئول شبکهٔ خودمختار مي‌تواند بر روي شبکهٔ تحت نظارت خود "حاکميّت" داشته باشد يعني مي‌تواند بر روي تك‌تك اجزاي شبکه ، طراحي زيرساخت ارتباطي و طريقهٔ اتصال شبکه‌هاي محلي و نوع پروتکل ، سيستم عامل(ماشينهاي ميزبان) ، توپولوژي کل شبکه مسيريابي اعمال نفوذ کرده و نظرات خود را پياده نمايد.

مسیریابی در شبکه های خود مختار

مسیریابي بسته‌هاي IP در درون يك شبكة خودمختار بيشتر تابع پارامترهايي نظير سرعت و قابل اعتماد بودن الگوريتم مسيريابي است .

دروازه‌های مرزی Border Gateway:

مسيريابهايي كه ارتباط دو شبكة خودمختار متفاوت را برقرار مي‌كنند و تمامي ارتباطات بين‌شبكه‌اي از طريق آنها انجام مي‌شود .

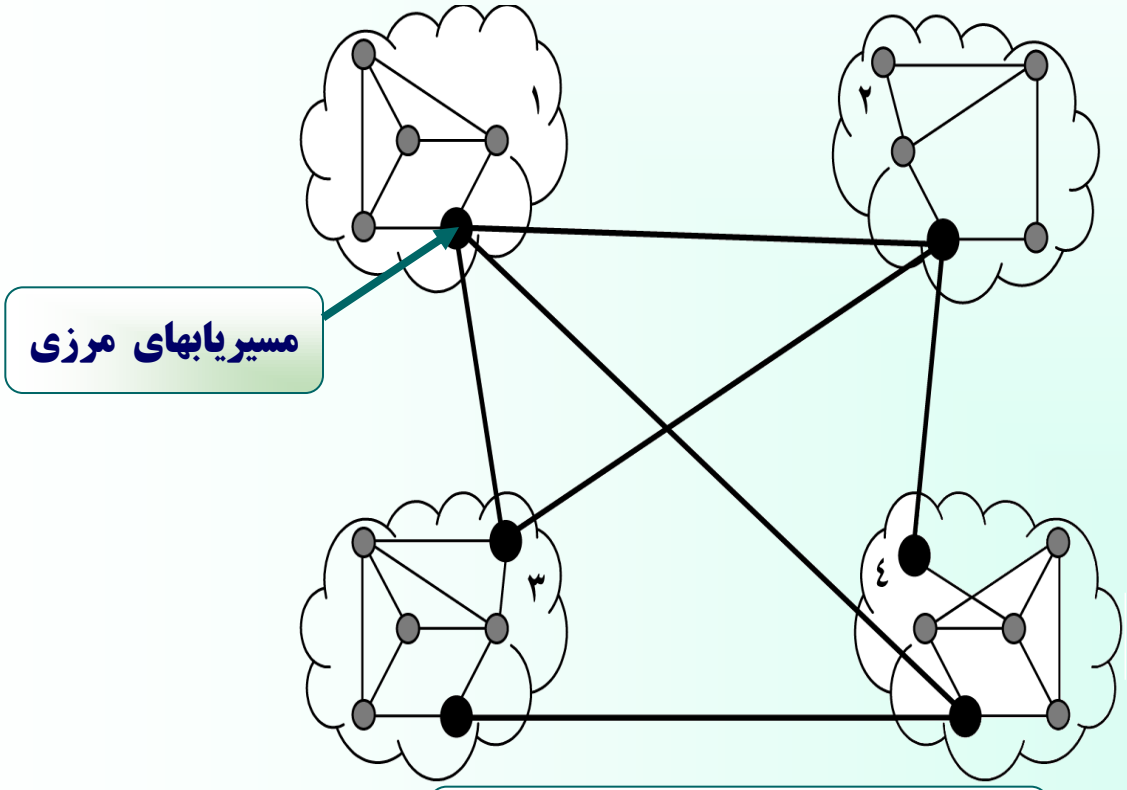
دروازه‌های مرزی Interior Gateway

مسيريابهايي كه ارتباط دو شبكة خودمختار متفاوت را برقرار مي‌كنند و تمامي ارتباطات بين‌شبكه‌اي از طريق آنها انجام مي‌شود.

- مسيريابهاي مرزي و ساختار ارتباطي بين آنها تابع قواعد "مسيريابي بروني"
- مسيريابهاي داخلي تابع الگوريتمهاي "مسيريابي دروني" مرزي
- مسيريابهاي مرزي = مسيريابهاي BGP

مثال: اگر يك ماشين ميزبان در شبكة 1 بخواهد بسته‌اي براي ماشين ديگر در شبكة 4 بفرستد سه مرحله مسيريابي لازم است:

- مسيريابي در درون شبكة 1 تا رسيدن بسته به مسيرياب مرزي
- مسيريابي روي خطوط ارتباطي بين‌شبكه‌اي تا رسيدن به شبكة 4
- مسيريابي درون شبكة 4 تا رسيدن به ماشين مقصد

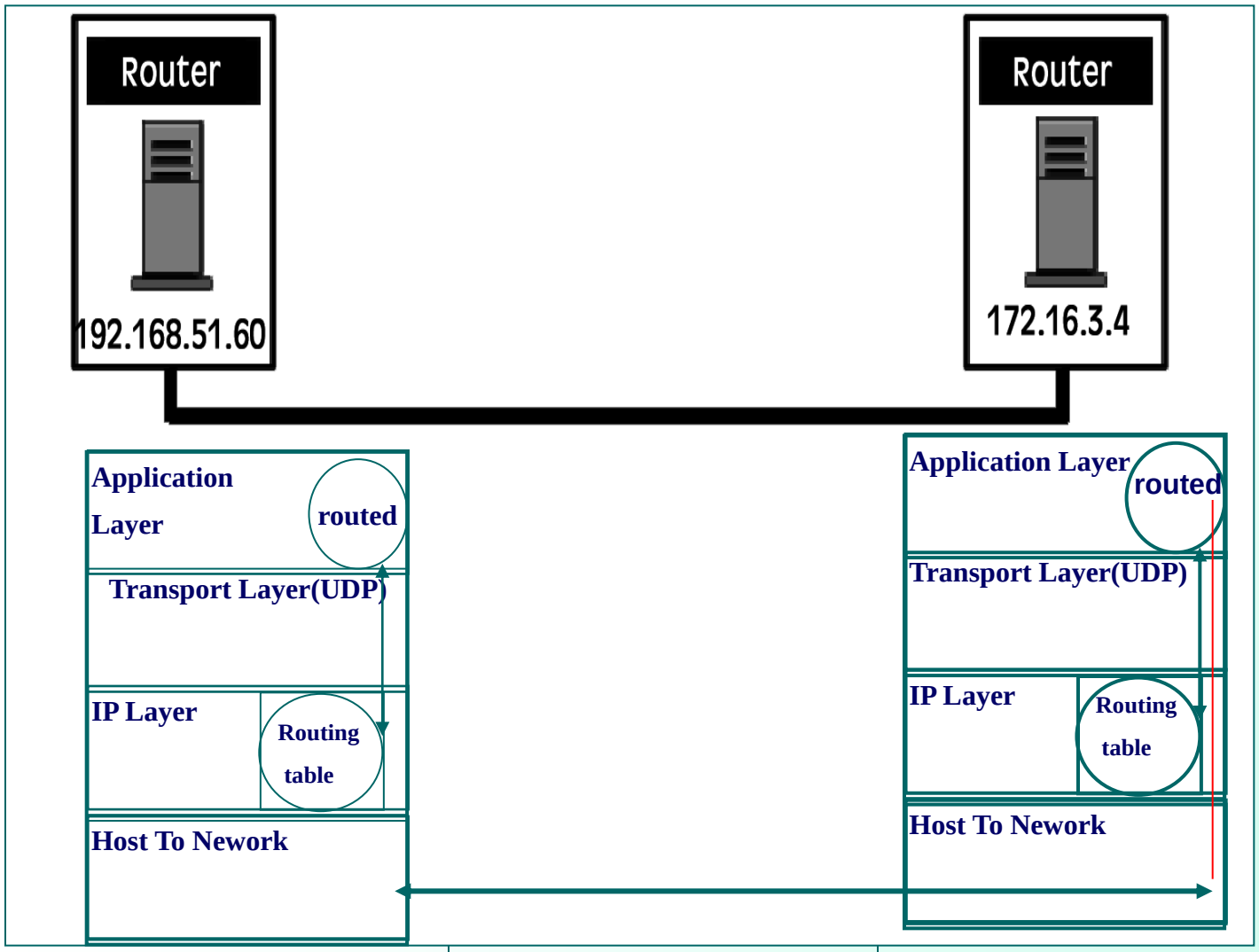


مثالی از چهار شبکهٔ AS متصل به هم

پروتکلRIP در مسیریابی درونی : Routing Information Protocol

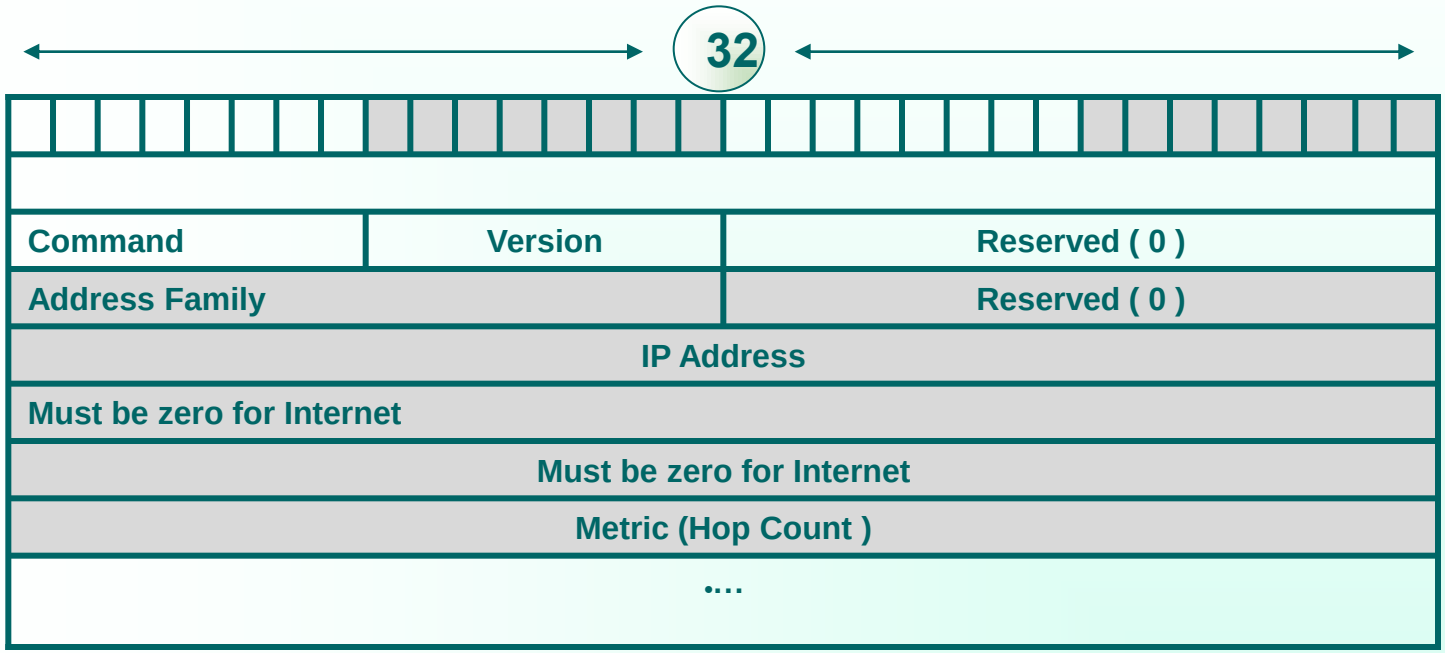
- اولين پروتكل مسيريابي دروني (1982)
- مبتني بر الگوريتم بردار فاصله DV
- معيار هزينه = تعداد گام
- مبادله جداول مسيريابي هر 30 ثانيه يكبار بين مسيريابهاي مجاور
- حداكثر تعداد طول مسير = 15
- استفاده از پروتكل UDP و پورت شماره 250 جهت مبادله جداول مسيريابي

جداول مسيريابي در لايه دوم جهت مسيريابي بسته‌هاي IP
مبادله جداول و عمليات به هنگام‌سازي توسط برنامه كاربردي لايه چهارم



پروتكل RIP در لاية كاربرد

قالب پیامها در پروتکل RIP



پروتکل OSPF در مسیریابی درونی Open Shortest Path First

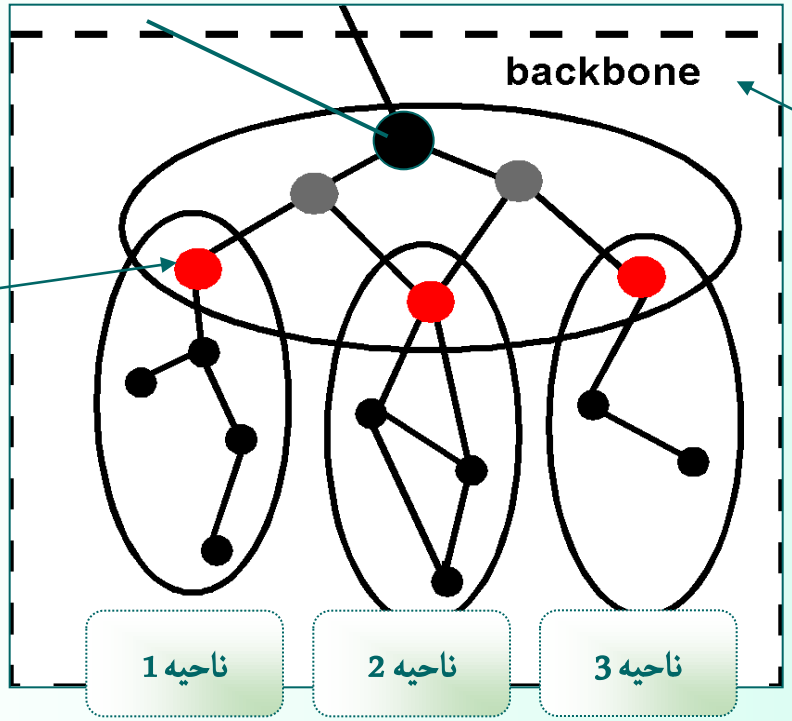
مقایسه پروتکل OSPF با RIP

- استفاده از الگوریتم LS براي محاسبة بهترين مسير بر خلاف پروتكل RIP و عدم وجود مشكل "شمارش تا بينهايت"
- توانايي در نظر گرفتن چندين معيار هزينه در انتخاب بهترين مسير برخلاف پروتكل RIP
- در نظرگرفتن حجم بار و ترافيك يك مسيرياب در محاسبة بهترين مسير بر خلاف پروتكل RIP و همگرايي سريع جداول مسيريابي در هنگام خرابي يك مسيرياب
- انتخاب مسير مناسب براي يك بسته بر اساس نوع سرويس درخواستي با توجه به فيلد Type of Service در بستة IP بر خلاف پروتكل RIP

مقايسه پروتكل OSPF با RIP

- هدايت نكردن تمام بسته‌هاي ارسالي براي يك مقصد خاص ، روي بهترين مسير و ارسال درصدي از بسته‌ها روي مسيرهاي در رتبه 2و 3 و ... از نظر هزينه ، بر خلاف پروتكل RIP = موازنه = Load Balancing
- پشتيباني از مسيريابي سلسله‌مراتبي برخلاف پروتكل RIP
- عدم قبول جداول مسيريابي مسيريابها توسط هر مسيرياب بدون احراز هويت ارسال‌كنندة آن
- استفاده مستقيم از پروتكل IP برخلاف پروتكل RIP (استفاده از پروتكل UDP در لايه انتقال)

- تقسيم يك شبكه خود مختار به تعدادي ناحيه و اطلاع تمام مسيريابهاي درون يك ناحيه از مسيريابهاي هم ناحيه و هزينه ارتباط بين آنها و ذخيره آن در جدول
- ارسال جداول براي تمام مسيريابهاي هم ناحيه در زمانهاي بهنگام‌سازي



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

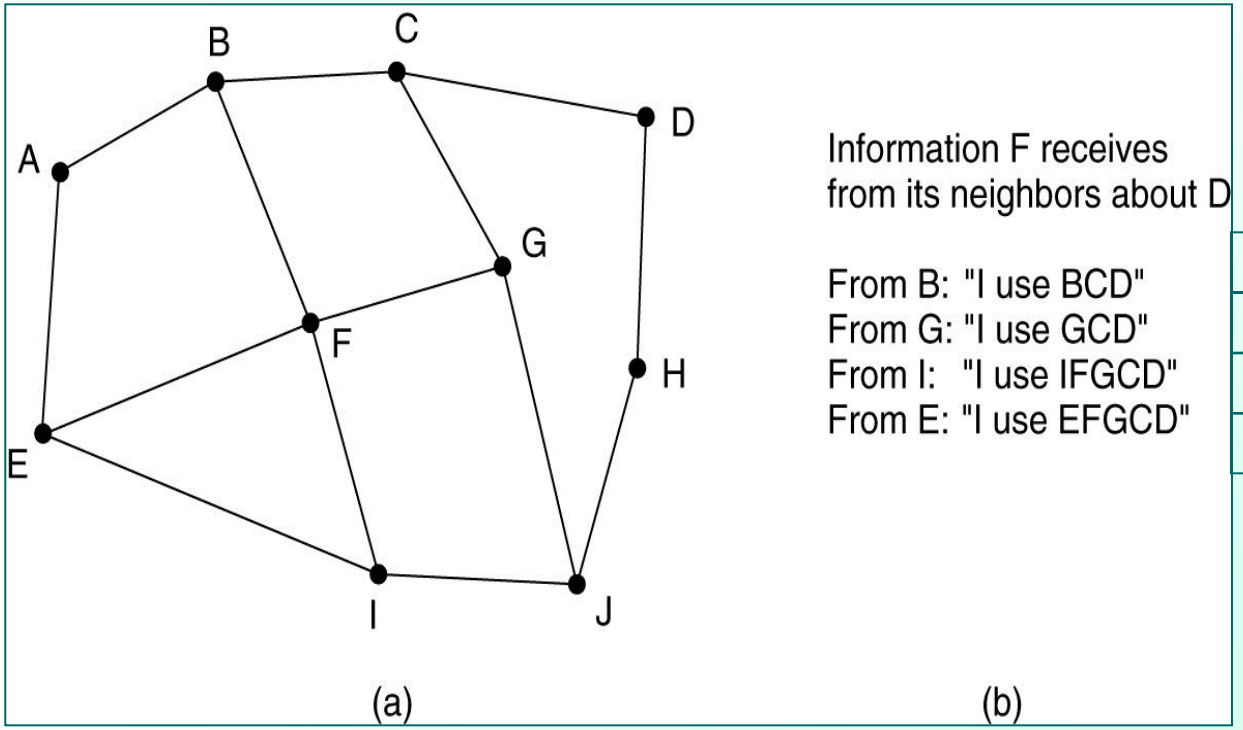
مسیریابهای مرزی
برقرارکننده ارتباط نواحی

سلسله‌مراتب مسیریابی در پروتکل OSPF

پروتکل BGP : پروتکل مسیریابی برونی The Exterior Gateway Routing Protocol

- الگوريتمهاي مسيريابي بين شبكه‌هاي خود مختار در اينترنت : BGP
- به جاي مبادله جداول مسيريابي و هزينه‌ها در پروتكل BGP بين مسيريابهاي مجاور ، ارسال فهرستي از مسيرهاي كامل بين هر دو مسيرياب در شبكه براي مسيريابهاي مجاور در بازه‌هاي زماني T ثانيه‌اي (بدون تعيين هزينه)

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



تعیین مسیر رسیده از B

تعیین مسیر رسیده از G

تعیین مسیر رسیده از I

تعیین مسیر رسیده از B

ساختار فرضي از ارتباط بين مسيريابهاي BGP

الگوريتمهائي كه در تبادل اطلاعات با همسايگان مسيرهاي كامل را به اطلاع يكديگر مي‌رسانند:

اولاً: مشكل "شمارش تا بينهايت" را نخواهد داشت. مانند پروتكل **BGP**

ثانياً: مسيريابهاي ديگر مي‌توانند بر روي كل مسير ، بررسي‌هاي امنيتي ، اقتصادي ، سياسي و ملي انجام دهند و بر اساس اين پارامترها مسير مناسب را انتخاب نمايند. مانند پروتكل **BGP**

تبادل اطلاعات مسيريابي (فهرست مسيرها) در پروتكل **BGP** در قالب پيام

انواع پيام تعريف شده در پروتكل **BGP**:

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

فصل سوم : لایه انتقال در شبکه اینترنت

هدفهای آموزشی :

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


پروتکلهای لایه انتقال
UDP
User Datagram Protocol
TCP
Transmisson Control Protocol

لايه IP

- هدايت و مسيريابي بسته‌هاي اطلاعاتي از يک ماشين ميزبان به ماشين ديگر
- عدم حل مشکلات احتمالي به وجود آمده براي بسته‌هاي IP در مسير

لايه انتقال

- فراهم آوردن خدمات سازماندهي‌شده, مبتني بر اصول سيستم عامل, براي برنامه‌هاي کاربردي در لايه بالاتر
- جبران کاستي‌هاي لايه IP

کاستی‌های لایه IP

- عدم تضمین درآماده‌بودن ماشین مقصد جهت دریافت بسته

O عدم تضمین در به ترتیب رسیدن بسته‌هاي متوالي و داده‌ها و صحت آنها

راهکارهای پروتکلTCP

- برقراري یک ارتباط و اقدام به هماهنگي بین مبدأ و مقصد قبل از ارسال هر گونه داده

O قراردادن شماره ترتیب براي داده‌ها

O تنظیم کد 16 بیتي کشف خطا در مبدأ و بررسي مجدد آن در مقصد جهت اطمینان از صحت داده‌ها

کاستی‌های لایه IP

❖ عدم تمایز در دریافت بسته‌هاي تکراري در مقصد (Duplication Problem)

➢ عدم تنظیم سرعت ارسال و تحویل بسته‌ها

❑ عدم توزیع بسته‌ها بین پروسه‌هاي مختلف اجرا شده بر روي یک ماشین واحد

راهکارهای پروتکل TCP

❖ قرار دادن شماره ترتیب در بسته ارسالي

➢ استفاده از الگوریتم پویا جهت تنظیم مجموعه زمانسنجها

❑ قراردادن آدرس پورت پروسه فرستنده و گیرنده در سرآیند بسته ارسالي

آدرس پورت

شماره شناسايي مشخص‌كننده هر پروسه براي برقراري يک ارتباط با پروسه‌ي ديگر بر روي شبکه

شماره پورتهای استاندارد

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

آدرس سوکت

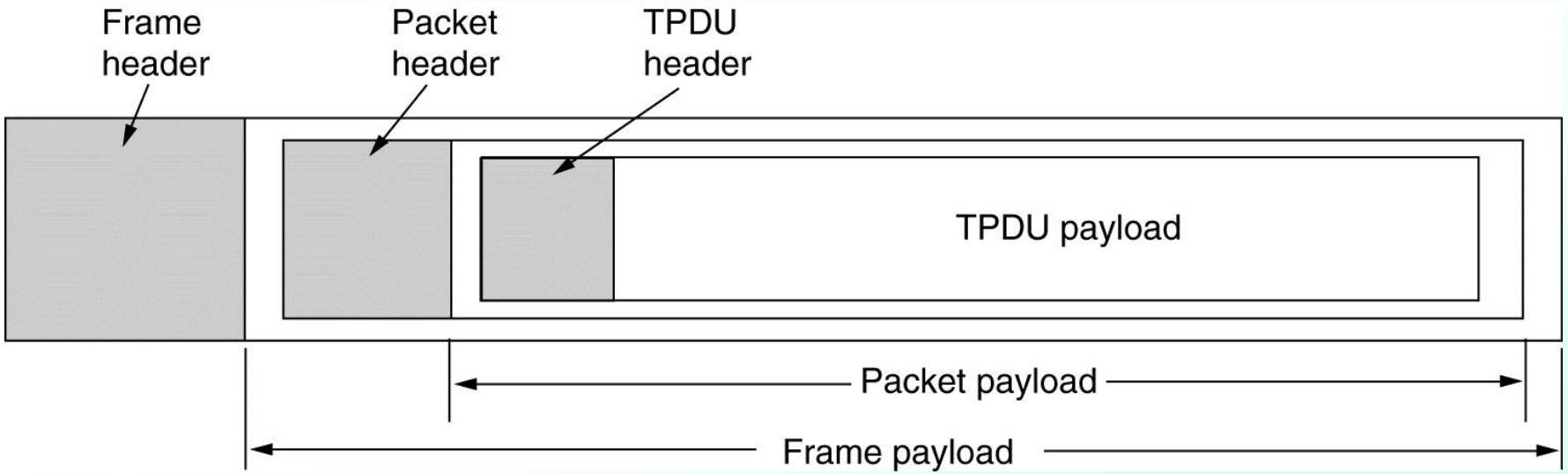
زوج آدرس IP و آدرس پورت مشخص‌کننده یک پروسه یکتا و واحد بر روي هر ماشین در دنیا

(IP Address: Port Number)= Socket Address

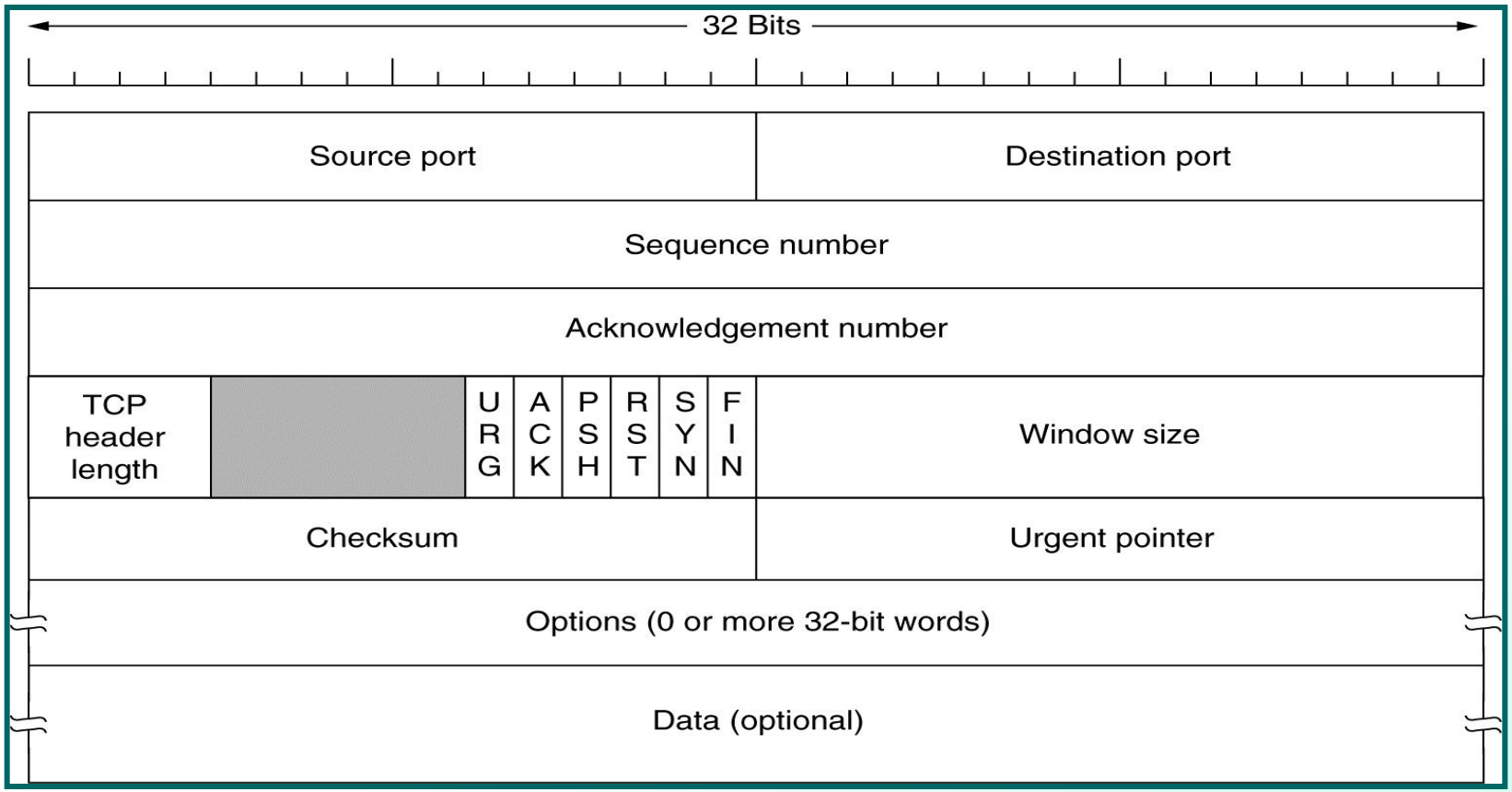
مثال 80 : 193.142.22.121

ساختار بسته های پروتکل TCP

TPDU = Transport Protocol Data Unit= بسته تولید شده در لایه انتقال = قطعهTCP



بسته پروتکل TCP



فیلد Source Port

- فیلد 16بیتي
- آدرس پورت پروسه مبدأ

فیلد Destination Port

- فیلد 16 بیتي
- آدرس پورت پروسه مقصد

فیلد Sequence Number

- فیلد 32 بیتي
- مشخص کننده شماره ترتیب آخرین بایت قرارگرفته شده در فیلد داده از بسته جاري

فیلد Acknowledgement Number

- فیلد 32 بیتي
- مشخص‌کننده شماره ترتیب بایتي که فرستنده بسته منتظر دریافت آن است

فیلد TCP Header Lenght

- فیلد 4 بیتي
- مشخص کننده طول سرآیند بسته TCP برمبناي کلمات 32 بیتي
- حداقل مقدار = 5
- تعیین کننده محل شروع داده‌ها در بسته TCP

۶ بیت بلااستفاده

6 بیت بلااستفاده جهت استفاده درآینده

بیتهای Flag

۶ بیتی

URG	ACK	PSH	RST	SYN	FIN

بیت URG

مقدار فیلد = 1 نشان دهنده معتبر بودن مقدار موجود در فیلد Urgent Pointer

مقدار فیلد = 0 نشان دهنده نا معتبربودن مقدار موجود در فیلد Urgent Pointer

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

بیت ACK

مقدار فیلد = 1 نشان‌دهنده معتبر بودن مقدار موجود در فیلد
Acknowledgement Number

بیت PSH (PUSH)

مقدار فیلد = 1 نشان‌دهنده تقاضاي فرستنده اطلاعات از گیرنده اطلاعات جهت بافرنکردن داده‌هاي موجود در بسته و تحویل سریع بسته به برنامه‌هاي کاربردي به منظور انجام پردازشهاي بعدي

بیت RST

مقدار فیلد = 1 نشان‌دهنده قطع ارتباط به صورت یکطرفه و ناهماهنگ

بیت SYN

تغییر مقدار این فیلد جهت برقراري ارتباط توسط ماشین

روند برقرای ارتباط TCP

الف) تنظیم بیتهاي ACK=0 و SYN=1 توسط شروع کننده ارتباط در یک بسته TCP بدون داده (تقاضاي برقراري ارتباط = Connection Request)

ب) تنظیم بیتهاي SYN=1 و ACK=1 در صورت قبول طرف دریافت‌کننده بسته تقاضاي برقراري ارتباط به برقراري ارتباط

بیت FIN

مشخص‌کننده قطع و پایان ارسال اطلاعات هنگام اتمام داده‌هاي ارسالي توسط طرفين با 1 نمودن مقدار اين بيت هنگام ارسال آخرين بسته

قطع كامل ارتباط: 1 نمودن مقدار اين فيلد توسط هر دو ماشين فرستنده و گيرنده

قطع ارتباط يكطرفه: 1 نمودن مقدار اين فيلد توسط يكي از طرفين ارتباط

فیلد Windows Size

مشخص كننده مقدار ظرفيت خالي فضاي بافر گيرنده

فیلد Checksum

- فیلد 16 بیتي
- حاوي کد کشف خطا

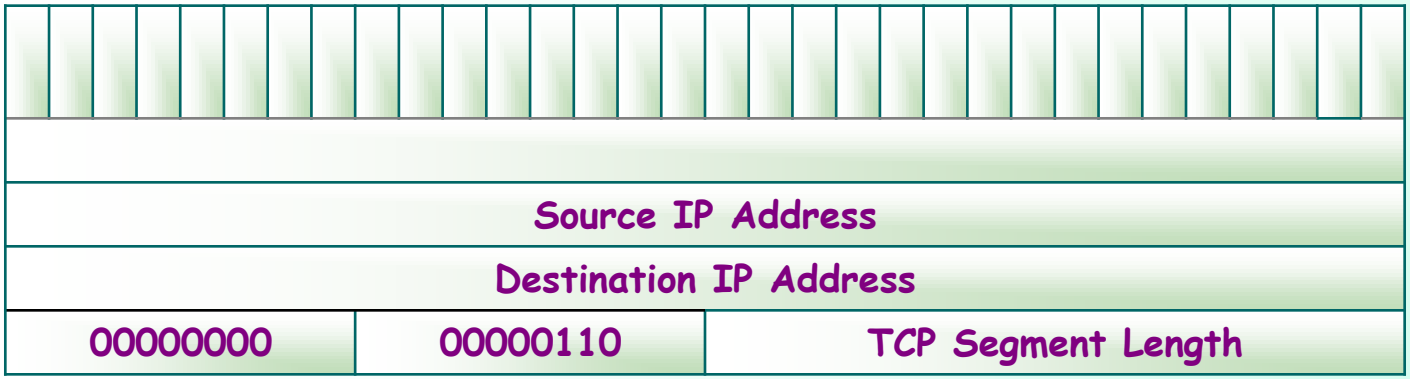
طریقه محاسبه کد کشف خطا

- تقسیم کل بسته TCP به قالبهاي 16 بیتي (منهاي قسمت Checksum)
- ایجاد یک سرآیند فرضي و تقسیم آن به صورت کلمات 16 بیتي
- جمع تمامي کلمات در مبناي مکمل 1 و منفي نمودن عدد حاصل در مبناي مکمل 1 و قرارگرفتن عدد حاصل در فیلد Checksum

جمع کل کلمات 16 بیتي موجود در بسته TCP + سرآیند فرضي = 0 عدم بروز خطا در حین ارسال داده‌ها

ساختار سرآیند فرضی

- 32 بیت آدرس IP ماشین مبدأ
- 32 بیت آدرس IP ماشین مقصد
- یک فیلد 8 بیتي کاملاً صفر
- فیلد 8 بیتي پروتکل که براي پروتکل TCP = 6
- فیلد TCP Segment Length = طول کل بسته TCP



فيلد Urgent Pointer

اشاره گر به موقعيت داده‌هاي اضطراري موجود در بسته TCP

فيلد Option

- فيلد اختياري
- شامل مقدار حداكثر طول بسته
- قراردادن كدهاي بي ارزش در اين فيلد به جهت آنكه طول بسته ضريبي از 4 باقي بماند

روش برقراری ارتباط در پروتکل TCP

روش دست تکاني سه مرحله‌اي

مرحله اول:

- ارسالِ یک بسته TCP خالي از داده از طرف شروع‌کننده ارتباط با بیتهاي SYN=1 و ACK=0 و قراردادن عدد x درون فیلد شماره ترتیب

- اعلام شروع ترتیب داده‌هاي ارسالي از x+1 به ماشین طرف مقابل
- پیشگیري از مساوي بودن شماره ترتیب داده‌هاي ارسالي با انتخاب مقدار X به صورت تصادفي

روش دست تكاني سه مرحله‌اي

مرحله دوم :

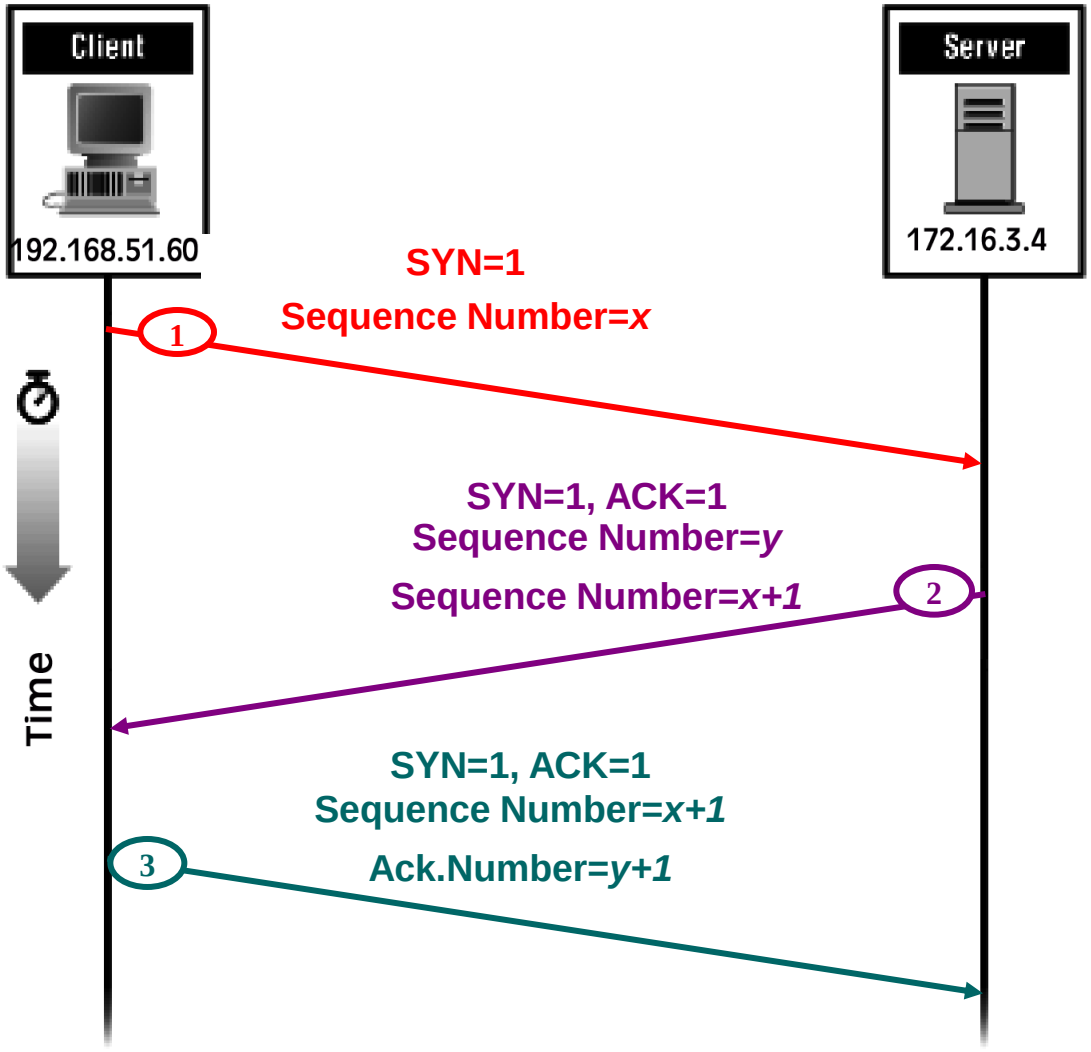
- **رد تقاضاي برقراري ارتباط:** ارسال بسته‌اي خالي با بيت **RST=1**
- **قبول تقاضاي برقراري ارتباط:** ارسال بسته خالي با مشخصات زير از طرف گيرنده بسته تقاضا:
  - بيت **SYN = 1**
  - بيت **ACK = 1**
  - **Acknowledgement = x+1**
  - **Sequence Number = y**

روش دست تكاني سه مرحله‌اي

مرحله سوم:

تصديق شروع ارتباط از طرف شروع‌كننده ارتباط با قراردادن مقادير زير در بيتهاي:

- SYN = 1
- ACK = 1
- Acknowledgement Number = y + 1
- Seq. No = x + 1



مراحل دست تکانی سه مرحله ای برای برقراری ارتباط در پروتکلTCP

روند خاتمه ارتباط TCP

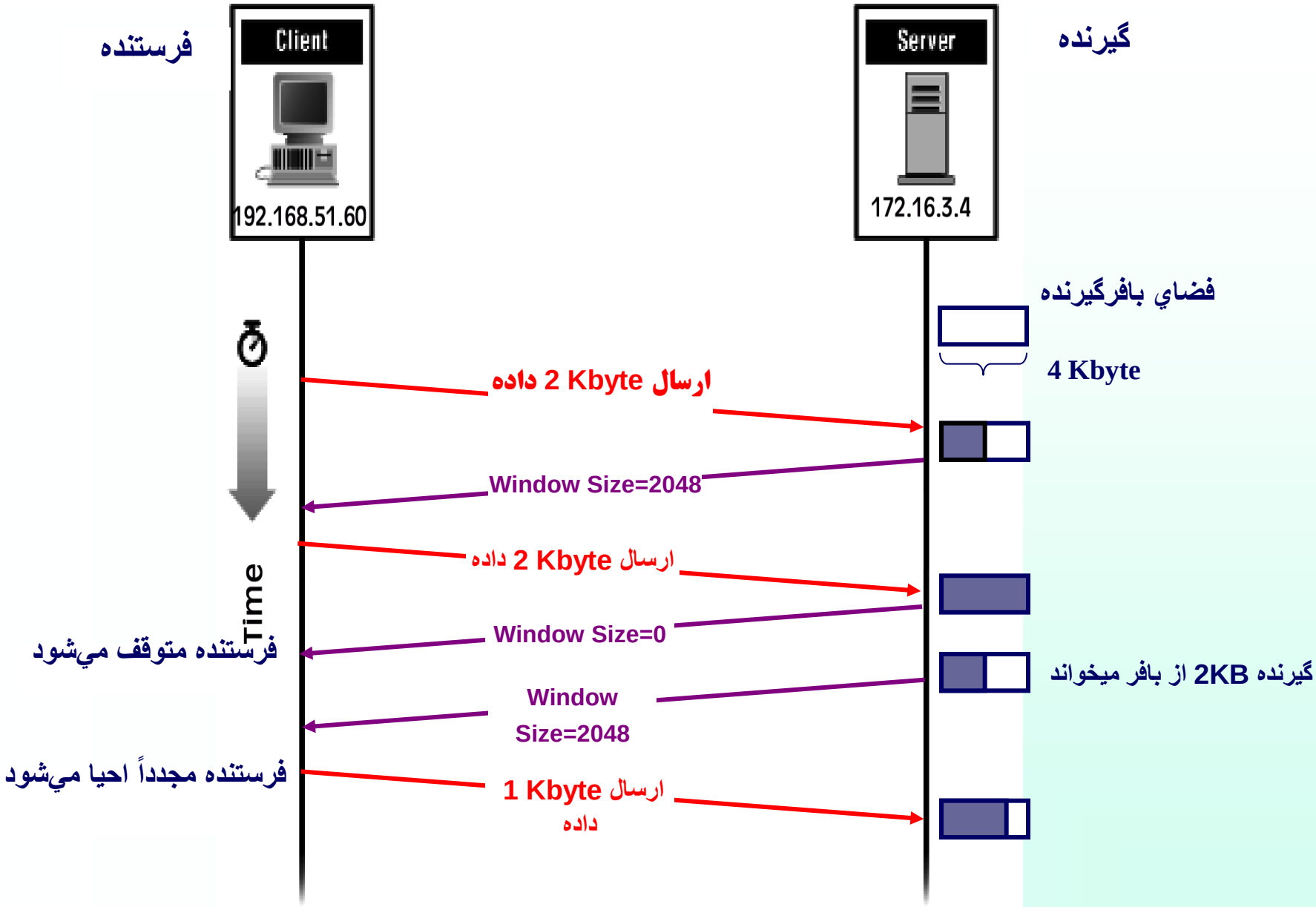
- ارسال بسته TCP با بیت FIN = 1 از طرف درخواست‌کننده اتمام ارسال
- موافقت طرف مقابل با اتمام ارتباط یکطرفه و ادامه ارسال داده توسط آن
- قطع ارتباط دو طرفه با یک نمودن مقدار بیت FIN در آخرین بسته ارسالي و تصدیق پایان ارتباط از طرف مقابل

کنترل جریان در پروتکل TCP

- استفاده از بافر جهت کنترل جریان داده‌ها در پروتکل TCP
- بافرشدن داده‌ها قبل از ارسال به برنامه کاربردي لايه بالاتر
- امکان عدم دریافت و ذخیره داده‌ها توسط برنامه کاربردي درمهلت مقرر و پرشدن بافر اعلام حجم فضاي آزاد بافر در فیلد Window درهنگام ارسال بسته TCP به طرف مقابل
- ایجاد یک ساختمان داده خاص به ازاي هر ارتباط برقرارشده TCP و نگهداري اطلاعاتي ازآخرين وضعيت ارسال و دريافت جریان داده‌ها = ساختمان داده بلوک نظارت بر انتقال = TCB = Transmission Control Block

نام متغير	توضيح
متغيرهای نظارت بر ارسال داده‌ها	
SND.UNA	شمارة ترتيب آخرين بسته اي كه ارسال شده ولي هنوز پيغام Ack آن برنگشته است.
SND.NXT	شمارة ترتيب آخرين بايت كه داده ها از آن شماره به بعد در بستة بعدي كه بايد ارسال شود.
SND.WND	ميزان فضاي آزاد در بافر ارسال
SND.UP	شمارة ترتيب آخرين داده هاي اضطراري كه تحويل برنامة كاربردي شده است.
SND .WL1	
SND.WL2	
SND.PUSH	شمارة ترتيب آخرين داده هايي كه بايد آني به برنامة كاربردي گسيل (Push) شود.
SND.ISS	مقدار اولية شمارندة ترتيب داده هاي دريافتي كه در حين ارتباط بر روي آن توافق مي‌شود.
متغيرهای نظارت بر دريافت داده‌ها	
RCV.NXT	شمارة ترتيب آخرين بايت در بستة بعدي كه از آن شماره به بعد انتظار دريافت آنرا دارد.
RCV.WND	ميزان فضاي آزاد در بافر دريافت
RCV.UP	شمارة ترتيب آخرين داده هاي اضطراري كه براي برنامة طرف مقابل ارسال شده است.
RCV.IRS	مقدار اولية شمارندة ترتيب داده هاي ارسالي كه در حين ارتباط بر روي آن توافق مي‌شود.

متغيرهاي ساختمان داده TCP



مثال روند کنترل جریان در پروتکل TCP

زمان سنجها در پروتکل TCP

TCP Timer

وابستگي عملكرد صحيح پروتكل **TCP** به استفاده درست از زمان سنجها

زمان سنجها

- ❑ Retransmission Timer
- ❑ Keep- Alive Timer
- ❑ Persistence Timer
- ❑ Quite Timer
- ❑ Idle Timer

زمان سنج Retransmission Timer

پس از برقراري ارتباط و ارسال بسته براي پروسه مقصد, زمان‌سنجي (RT) با مقدار پيش فرض تنظيم و فعال مي‌گردد و شروع به شمارش معكوس مي‌نمايد كه اگر در مهلت مقرر پيغام دريافت بسته (Ack) نرسيد رخداد انقضاي زمان تكرار روي داده و ارسال مجدد بسته صورت گيرد.

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

1- عمل ارسال مجدد یک بسته چند بار باید تکرارشود ؟

2- مقدار پیش فرض زمان سنج چه مقدار باشد ؟

بهترین راه تنظیم زمان سنج : روشهاي وفقي و پويا

الف) ایجاد یک متغیر حافظه یه نام RTT و مقداردهي آن هنگام برقراري یک ارتباط TCP

ب) تنظیم یک زمان‌سنج به ازاي ارسال هر بسته و اندازه زمان رفت و برگشت پیغام دریافت بسته = M

الگوریتم Jacobson

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

$$RTT_{new}=RTT_{old}+4*D_{new}$$

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

$$\alpha=7/8$$

مقدار اولیه D مي‌تواند صفر باشد.

Keep- Alive Timer

- توقف ارسال اطلاعات و عدم تبادل داده علي رغم فعال و باز بودن ارتباط TCP
- قطع ارتباط يكي از طرفين به دليل خرابي سخت افزاري و يا نرم افزاري

جهت تمايز اين دو حالت

ارسال بسته TCP خالي از داده از طرف فرستنده اطلاعات براي مقصد با استفاده از زمان سنج Keep-Alive Timer (زمان پيش فرض بين 5 تا 45 ثانيه)

عدم بازگشت پيغام دريافت

قطع ارتباط به صورت يكطرفه و آزاد نمودن تمام بافرها

بازگشت پيغام دريافت از طرف مقصد

ارتباط TCP باز و فعال است

- مقدار فضاي بافر آزاد يكي از طرفين ارتباط صفر (Window Size= 0) متوقف شدن پروسه طرف مقابل
- خالي شدن مقداري از فضاي بافر پر شده بعد از مدتي اعلام آزادشدن فضاي بافر جهت احياي پروسه بلوكه و متوقف شده توسط سيستم عامل و شروع و ادامه ارسال پروسه متوقف شده

Persistence Timer

ارسال بسته TCP در فواصل زماني منظم با استفاده از زمان سنج Persistence Timer پس از آزاد شدن فضاي بافر براي پروسه بلوكه‌شده جهت احيا و ادامه ارسال داده توسط آن

Quite Timer

هنگام بسته شدن يک ارتباط TCP با شماره پورت خاص تا مدت زمان معيني که زمان سنج Quite Timer تعيين مي نمايد (مقدار پيش فرض = 30 تا 120 ثانيه) هيچ پروسه اي اجازه استفاده از شماره پورت بسته شده را ندارد.

جهت رسيدن بسته هاي سرگردان ناشي از ارتباط پايان يافته موجود در شبکه به مقصد

Idle Timer

اگر تلاش براي تکرار ارسال يک بسته بيش از حد متعارف انجام شود ارتباط TCP را بصورت يکطرفه رها کرده و قطع مي‌نمايد. مقدارمعمول آن 360 ثانيه است.