禪

آموزش مفاهیم شبکه به زبان ساده

آشنایی با پروتکلها و مسیریابی شبکه _

TCP/IP

در این مقاله قصد داریم درباره نحوه عملکرد پروتکلها در لایههای مختلف و پیامهایی که ایجاد میکنند اطلاعات بیشتری به دست آورده، با نحوه کار روترها آشنا شده و نیم نگاهی به لایه شبکه داشته باشیم .

این مطلب بخشی از سری آموزشهای نتورک پلاس است که پیشتر در سایت شبکه منتشر شده است.
برای مشاهده فهرست و خرید کتاب +Network راهنمای شبکهها اینجا کلیک کنید .

در مقالههای آغازین آموزش نتورکپلاس به شما گفتیم یک پروتکل نقش حاکمیتی داشته و به کامپیوترهای تحت یک شبکه اعلام میکند که دستورالعملها و دادهها را بر مبنای چه روشی انتقال دهند، در ادامه با تجهیزات زیرساختی شبکه آشنا شدیم و اطلاعاتی درباره نحوه کار لایههای شبکه، کاربرد، انتقال و پیوند داده به دست آورده و خواندیم که لایهها چگونه به تجهیزات شبکه کمک میکنند انواع مختلفی از آدرسها را برای ارسال درست دادهها برای دستگاهها یا پردازهها به خدمت گیرند . در ادامه با وظایف لایههای مختلف مدل OSI همچون قالببندی، آدرسدهی و کشف خطاها اطلاعاتی آشنا شدیم. در بطن همه این فعالیتها و اتفاقاتی که به آنها اشاره داشتیم پروتکلها قرار دارند . اکنون قصد داریم درباره نحوه عملکرد پروتکلها در لایههای مختلف و پیامهایی که ایجاد میکنند اطلاعات بیشتری به دست آورده، با نحوه کار روترها آشنا شده و نیم نگاهی به لایه شبکه داشته باشیم .

پروتکلهای اصلی TCP/IP

TCP/IP را مجموعهای از پروتکلها یا استانداردهایی همچون TCP، IP(IPv4/Ipv6)، DUP، ARP به انضمام سایر پروتکلها به وجود آوردهاند . اما چه اتفاقی برای پیامهای سرآیند در لایه انتقال رخ میدهد؟

هر لایه داده ها و آدرس های خود را برای انتقال اطلاعات به سمت دستگاه موردنظر به لایه های متناظر اضافه می کند

7
Application
6
Presentation
5
Session
4
Transport
3
Network
2
Data Link
1
Physical
Firewall
Router
Switch
Payload
Header
Payload
Segment or datagram
Header
Header
Payload
Packet
Header
Header
Header
Payload
Trailer
Frame

پروتکل کنترل انتقال (TCP) سرنام **Transmission Control Protocol**

اگر به خاطر داشته باشید، به شما گفتیم پروتکل TCP در لایه انتقال از مدل OSI استفاده میشود و سرویسهایی قابل اعتماد برای تحویل دادهها ارائه میکند . اجازه دهید برای روشن شدن مطلب پروتکل TCP را با یک تماس تلفنی مقایسه کنیم تا سه ویژگی کاربردی TCP که ضمنات میدهند دادهها به شکل درستی انتقال پیدا میکنند را به خوبی درک کنیم .

اتصالگرایی - قبل از آنکه TCP دادهها را انتقال دهد، این پروتکل مطمئن میشود که یک ارتباط یا نشستی ایجاد شده است. شبیه به حالتی که یک تماس تلفنی برقرار میکنید و پیش از آنکه صحبتهای خود را شروع کنید با گفتن سلام مطمئن میشود در آن سوی خط فردی در حال گوش دادن است. TCP از یک فرآیند سه مرحلهای که به نام دستدادن سه طرفه از آن نام برده میشود برای برقراری یک ارتباط TCP استفاده میکند . TCP تنها پس از آنکه ارتباطی را برقرار کرد، در ادامه فرآیند انتقال واقعی دادهها را آغاز میکند . بهطور مثال پاسخگویی به محاورهای که درباره دسترسی به یک صفحه وب مبتنی بر HTTP ارسال شده است.

توالی و بررسیها - زمانی که یک تماس تلفنی برقرار میکنید، ممکن است از شخص مقابل سوال کنید که آیا صدای شما را به درستی میشنوند و در صورت لزوم جملهای به زبان میآورید . مشابه همین حالت در دنیای شبکه رخ میدهد، TCP یک رشته کاراکتری که بررسیکنندهها (checksum) نام دارند را ارسال میکند . در طرف دیگر پروتکل TCP میزبان رشته مشابهی را تولید میکند . اگر فرآیند تطابق دو رشته با شکست روبرو شود، میزبان از مبدا درخواست میکند دادهها را دومرتبه ارسال کند . علاوه بر این، به دلیل اینکه پیامها همیشه به همان ترتیبی که ساخته شدهاند به مقصد نمیرسند، TCP یک شماره ترتیب زمانی را برای مشخص کردن هر سگمنت برای میزبان ارسال میکند. اگر ضرورتی داشته باشد، در مقصد سگمنتهایی که دریافت شدهاند دومرتبه مرتب میشوند .

کنترل جریان - در آن سوی خط اگر فردی که با او صحبت میکنید، به درستی متوجه حرفهای شما نشود، مجبور هستید آهنگ صحبت کردن خود را کند کنید تا واژهها به درستی شنیده شوند . مشابه چنین حالتی در دنیای شبکه کنترل جریان نام دارد . کنترل جریان فرآیندی است که ضمانت میکند پیامها با نرخ درستی در حال انتقال هستند . نرخ انتقال دادهها بر مبنای سرعتی که گیرنده قادر به دریافت دادهها است تنظیم میشود. بهطور مثال، فرض کنید که دریافتکننده اعلام میدارد که بافر او قادر به مدیریت 4000 بایت است. در این حالت فرستنده فرآیند انتقال را بر مبنای 4000 بایت تنظیم کرده و به انتقال یک یا چند بسته کوتاه همراه با حالت توقف ادامه داده و پیش از فرستاندن بستههای بعدی کمی مکث میکند تا دادهها به درستی انتقال پیدا کنند . TCP همه عناصر دخیل در این فرآیند همچون دست دادن سه طرفه، بررسیکنندهها، توالی و کنترل جریان را با ارسال دادهها به فیلدهایی در سرآیند TCP در ابتدای یک سگمنت TCP مدیریت میکند.

فیلدهای درون یک سگمنت TCP

شکل زیر فهرستی از آیتمهایی را نشان میدهد که فیلد نامیده شده و درون یک سگمنت TCP قرار دارند . هر بلوک نشان داده شده در شکل بیانگر مجموعهای از بیتها بوده و هر سطر بیانگر 32 بیت است. تصویر زیر یک سگمنت TCP را نشان میدهد که همه فیلدها به جزء آخرین مورد فیلد دادهای بوده و بخشی از سرآیند TCP هستند. محتوای فیلد دادهای پیامی است که لایه بالای لایه انتقال آنرا ارسال کرده است .

شماره بیت: 0
16
31
سرآیند TCP
Source port
Destination port
Sequence number
Acknowledgment
Header length
Reserved
URG
ACK
PSH
RST
SYN
FIN
Sliding-window size
Checksum
Urgent pointer
Options
Padding
Data

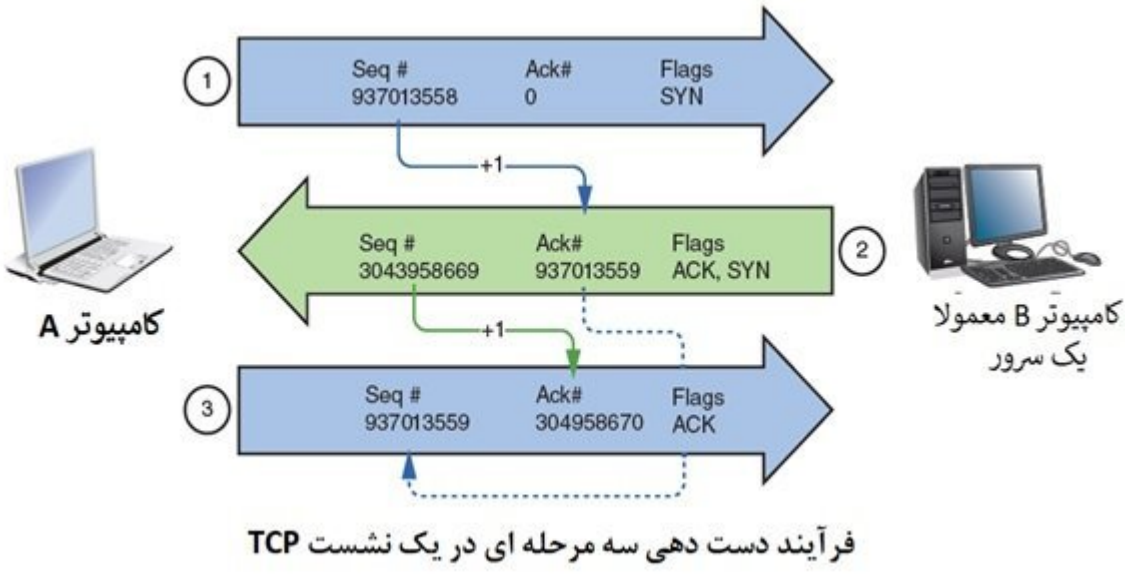
یک سگمنت TCP

نکته : سرآیندها در گروههای 32 ساخته میشوند که به آنها words گفته میشود. هر word (کلمه) شامل 4 بایت است که بلوک نامیده شده و هر کدام 8 بیت هستند. توضیح فوق به این دلیل آورده شد که سرآیندهای نشان داده شده در تصویر بالا همگی در گروههای 32 بیتی نشان داده شدهاند . توضیح هر یک از فیلدهای تصویر بالا در جدول زیر ارائه شده است . دقت کنید فیلد دادهای در انتهای تصویر بخشی از سرآیند TCP نیست. زمانی که سگمنت TCP به سمت پایین یعنی لایه شبکه میرود، کل سگمنت به بخش دادهای یک سگمنت آیپی تبدیل میشود . در ادامه این بارداده درون یک بسته آیپی کپسوله میشود .

	Source port	bits 16	[illegible]
	Destination port	bits 16	[illegible]
	Sequence number	bits 16	[illegible]
	Acknowledgment number	bits 32	[illegible]
	TCP header length	bits 4	[illegible]
	Reserved	bits 6	[illegible]
[illegible]	Flags	bits 6	[illegible] URG: [illegible] Urgent [illegible] ACK: [illegible] Acknowledgment [illegible] PSH: [illegible] RST: [illegible] SYN: [illegible] FIN: [illegible]
	Sliding-window (size (or window	bits 16	[illegible]
	Checksum	bits 16	[illegible]
	Urgent pointer	bits 16	[illegible]
	Options	32–0 bits	[illegible]
	Padding	[illegible]	[illegible]
	[illegible]	[illegible]	[illegible]

TCP در ارتباط دو کامپیوتر

برای برقراری ارتباط اطمینان پذیر TCP دو کامپیوتر/دستگاه باید قبل از شروع ارسال داده با یکدیگر هماهنگ شوند. این فرایند هماهنگی را دست دهی سه مرحله ای می نامند. این فرایند هماهنگی باعث می شود که هر دو کامپیوتر از آمادگی یکدیگر برای ارسال و دریافت داده مطمئن شوند. این فرایند هماهنگی در شکل زیر نشان داده شده است.



فرآیند دست دهی سه مرحله ای در یک نشست TCP

مراحل فرایند دست دهی سه مرحله ای را می توان به شکل زیر خلاصه کرد:

مرحله 1: کامپیوتر A یک بسته (قطعه) SYN ارسال می کند تا به کامپیوتر B اعلام کند که می خواهد ارتباط برقرار کند.

- در فیلد Sequence number کامپیوتر A عددی را قرار می دهد که برای شروع شماره گذاری بایت های داده ارسالی خود استفاده خواهد کرد. در این مثال این عدد 937013558 است.
- بیت SYN در 1 قرار داده می شود تا نشان دهد که این بسته برای برقراری ارتباط است. داده ای ارسال نمی شود. کامپیوتر A نیز می تواند برخی از گزینه های TCP را برای استفاده در ارتباط مشخص کند.
- بیت ACK در این بسته روشن نیست زیرا فیلد Acknowledgment number در این بسته 0 است زیرا کامپیوتر B هنوز چیزی برای تایید دریافت ارسال نکرده است.

مرحله 2: SYN/ACK (بسته ای که هم برای تایید و هم برای ارسال است)- کامپیوتر B این بسته را برای کامپیوتر A ارسال می کند تا به آن اعلام کند که:

- بیت های ACK و SYN هر دو 1 هستند. کلمه کلیدی است. کلمه کلیدی به معنای: "بخش ارسالی از من را دریافت کردم و من هم می خواهم ارتباط برقرار کنم."
- فیلد Acknowledgment number کامپیوتر A را مطلع می کند که کامپیوتر B تا چه بایتی را دریافت کرده است. (عدد 1). کامپیوتر B باید عدد 937013559 را به عدد دریافتی اضافه کند. این عدد 937013559 است. کامپیوتر B می خواهد به کامپیوتر A اعلام کند که بایت 937013559 را دریافت کرده است و منتظر بایت بعدی است.
- در فیلد Sequence number کامپیوتر B عددی را قرار می دهد که کامپیوتر A باید برای شماره گذاری بایت های داده ارسالی خود استفاده کند. در این مثال این عدد 3043959669 است.

مرحله 3: ACK (آخرین مرحله)- کامپیوتر A به کامپیوتر B اعلام می کند که بسته قبلی را دریافت کرده است.

- Sequence number 937013559 B .
- Acknowledgment number B 1 . 3043959670 .
- A ACK 1 . A .

A . 1 . A . (HTTP) A . A . 937014284 . 725 . 937013559 . 725 . FIN A 1 .

User Datagram Protocol (UDP)

UDP . TCP . UDP . UDP . UD

Bit number: 0
16
31
UDP header
Source port
Destination port
Length
Checksum
Data

یک دیتاگرام UDP

پروتکل IP سرنام Internet Protocol

پروتکل IP به لایه شبکه در مدل OSI تعلق دارد . این پروتکل مشخص میکند که دادهها به چه مکانی باید تحویل داده شده و همچنین آدرسهای آیپی مبدا و مقصد را مشخص میکند. IP پروتکلی است که TCP/IP را به شبکه متصل میکند. به عبارت دقیقتر به این پروتکل اجازه میدهد از میان شبکههای محلی مختلف با اتکا بر روترها عبور کند . همانطور که پیشتر گفتیم، در لایه شبکه از مدل OSI، دادهها درون بستههایی سازماندهی میشوند. یک بسته آیپی شامل اطلاعات مهمی است که روترها برای انتقال دادهها میان سگمنتهای مختلف شبکههای محلی به آنها نیاز دارند . IP یک پروتکل بدون اتصال است، به این معنی که IP نشستی برای ارسال بستههای خود منتشر نمیکند . هر بسته آیپی بهطور جداگانه از سایر بستههایی که درون مجموعه خودش قرار دارد ارسال میشود، در نتیجه برخی از پیامها ممکن است از مسیرهایی متفاوت از دیگری برای رسیدن به مقصد استفاده کنند . تایید این مسئله که آیا آیپی پیامی را به میزبان درستی تحویل داده است یا خیر بر عهده پروتکل TCP است. IP با اتکا بر پروتکل TCP یا UDP مطمئن میشود که هر پیام به برنامه درستی که روی میزبانی در حال اجرا است، تحویل داده شده است . همانگونه که عنوان شد، دو نسخه از پروتکل IP روی شبکههای امروزی استفاده میشود . IPv4 که اولین بار در سال 1981 معرفی شد و هنوز هم به عنوان استانداردی روی بیشتر شبکهها استفاده میشود و IPv6 که در سال 1998 معرفی شد که امنیت بهتر، تنظیمات اولویتبندی بهتر، تنظیمات پیکربندی خودکار بهتر و آدرسهای آیپی اضافیتر را ارائه میکند . بیشتر برنامهها، سرورها، کلاینتها و دستگاههای تحت شبکه از IPv6 پشتیبانی میکنند . با این حال، هزینه ارتقا زیرساختها به IPv6 برای بسیاری از سازمانها سنگین بوده و در نتیجه بیشتر سازمانها ترجیح میدهند از IPv4 استفاده کنند . به عنوان یک تکنسین شبکه، شما باید اطلاعات کافی در مورد هر دو نسخه این پروتکل به دست آورید . ابتدا اجازه دهید ببینیم بستههای IPv4 چگونه ساخته شده و پس از آن به سراغ بستههای IPv6 برویم .

بستههای IPv4

شکل زیر یک بسته IPv4 را نشان میدهد .

Bit number: 0
16
31
IP header
Version
IHL
DiffServ
Total length
Identification
Flags
Fragment offset
TTL
Protocol
Header checksum
Source IP address
Destination IP address
Options
Padding
Data (TCP segment or UDP datagram)

.IPv4 [illegible]

عملکرد	طول	فیلد	
نسخه پروتکل IP را مشخص میکند. بهطور مثال IPv4 یا Ipv6. یک ایستگاه کاری به فیلد فوق نگاه کرده تا بررسی کند که آیا میتواند دادههای وارد شونده را بخواند یا خیر . اگر موفق نشود بسته را برگشت میکند.	bits 4	Version	سرآیند
اندازه سرآیند آیپی را در واحد بایتها نشان میدهد . این سرآیند میتواند حداقل 20 بایت و حداکثر 60 بایت باشد . این فیلد همچنین Data offset نیز نامیده میشود، زیرا افست شروع بسته را تاوقتی که دادهها از سوی بسته حمل شوند را مشخص میکند.	bits 4	IHL (Internet header length)	
برای روترها سطح اولویتبندی بستههایی که قرار است پردازش شوند را مشخص میکنند.	bits 8	DiffServ (Differentiated services)	
طول کل بسته آیپی را در واحد بایت مشخص کرده و شامل سرآیند و داده است. یک بسته آیپی شامل سرآیند و داده بوده و اندازه آن نباید از 65353 بایت تجاوز کند .	bits 16	Total length	
برخی مواردروترها و میزبانهای مجبور به شکستن یک دیتاگرام به بستههای کوچکتر هستند. در این حالت ماشین مقصد مجبور به بازسازی بستهها است. زمانی که یک دیتاگرام واحد شکسته میشود، باید ویژگی وجود داشته باشد تا مقصد بتواند بستههای دریافتی را بازسازی کرده و آنها را از میان سایر بستههای دیتاگرام جدا کند . این فیلد و دو فیلد بعد Flags و Fragment offset به بازسازی بستههایی که جدا دریافت شدهاند کمک میکنند.	bits 16	Identification	
مشخص میکند که آیا یک یام شکسته شده و اگر شکسته شده است، آیا بستهای که دریافت شده آخرین قطعه شکسته شده است یا خیر . اولین بیت برای استفاده در آینده رزرو شده است .	bits 3	Flags	
مشخص میکند که بسته شکسته شده به چه مکانی در یک مجموعه وارد شده تعلق دارد .	bits 13	Fragment offset	
حداکثر مدت زمانی را مشخص میکند که یک بسته میتواند روی یک شبکه پیش از آنکه از دست برود باقی بماند . درست است که این فیلد واحدی از زمان را نشان میدهد، اما روی شبکههای مدرن این فیلد تعداد فدعاتی که یک بسته میتواند از طریق یک روتر فوروارد شده یا حداکثر تعداد دفعاتی که بسته از هر روتر میتواند عبور کند را نشان میدهد. مقدار TTL برای هر بسته متفاوت بوده و قابل پیکربندی است. بهطور معمول این مقدار به 32 یا 64 تنظیم میشود. هر بار که بستهای از یک روتر عبور میکند، TTL یک واحد کاهش پیدا میکند . زمانی که یک روتر یک بسته را با TTL عادل با 0 دریافت میکند، آنرا بسته را حذف کرده و یک پیام اتمام زمان پیام TTL را از طریق پروتکل ICMP برای مبدا ارسال میکند .	bits 8	TTL (Time to Live)	
نوع پروتکلی که بسته را دریافت میکند را مشخص میکند . (بهطور مثال UDP، TCP یا ICMP)	bits 8	Protocol	
به میزبان دریافت کننده بسته اجازه میدهد تا محاسبه کند که آیا سرآیند آیپی در هنگام دریافت بسته خراب شده است یا خیر . اگر فرآیند تطابق و ارزیابی وضعیت بستههای دریافتی درست نباشد به معنای آن است که بسته از دست رفته است.	bits 16	Header checksum	
آدرس آیپی مبدا را مشخص میکند.	bits 32	Source IP address	
آدرس آیپی مقصد را مشخص میکد.	bits 32	Destination IP address	
شامل اطلاعات زمانی و مسیریابی اختیاری است .	Variable	Options	
شامل بیتهایی است که اطمینان میدهند که سرآیند دارای بیتهای 32 است.	Variable	Padding	
شامل دادههایی است که اساسا از طرف مبدا ارسال شدهاند و همچنین شامل هر سرآیندی است که از لایههای بالاتر دریافت شدهاند . فیلد دادهای بخشی از سرآیند آیپی نیست و درون سرآیند آیپی کپسوله میشود.	Variable	Data	

نویسنده :
حمیدرضا_تائبی
دسته بندی:
فناوری شبکه
تاریخ انتشار :
20:20 - 10/08/1398
برچسب:
پروتکل شبکه - مسیریابی شبکه - آموزش_شبکه

نشانی
□□□□:https://www.shabakeh-mag.com/networking-technology/16171/%D8%A2%D8%B4%D9%86%D8%A7%DB%8C%DB%8C-%D8%A8%D8%A7-%D9%BE%D8%B1%D9%88%D8%AA%DA%A9%D9%84%E2%80%8C%D9%87%D8%A7-%D9%88-%D9%85%D8%B3%DB%8C%D8%B1%DB%8C%D8%A7%D8%A8%DB%8C-%D8%B4%D8%A8%DA%A9%D9%87