سپهر سلطانلو

Tcp.udpتفاوت

**TCP چیست؟**

پروتکل کنترل انتقال یا TCP، یک پروتکل connection-oriented است که کامپیوترها برای برقراری ارتباط از طریق اینترنت، از آن استفاده می کنند. یکی از اصلی ترین پروتکل ها، در شبکه های TCP/IP است. TCP، امکان بررسی خطا، تضمین تحویل داده ها و همچنین تحویل packetها به همان روشی که ارسال می شوند را فراهم می کند.

**UDP چیست؟**

User Datagram Protocol (UDP)، یک پروتکل connectionless است که درست مانند TCP کار می کند اما فرض می کند بررسی خطاها و خدمات recovery نیاز نیستند. در عوض، UDP به صورت مداوم، datagramها را برای کاربران ارسال می کند تا مطمئن شود که آن ها را دریافت کرده اند.

**تفاوت TCP و UDP**

تفاوت ها و شباهت های زیادی بین [**TCP و  UDP**](https://www.privateinternetaccess.com/blog/tcp-vs-udp-understanding-the-difference/#:~:text=TCP%20is%20a%20connection%2Doriented,a%20connection%20before%20sending%20data.) وجود دارد. هر دو متداول ترین روش برای ارسال packet از طریق اینترنت هستند و هر دو آن ها روی لایه Transport layer پروتکل کار می کنند، در ضمن هر دو از [**پروتکل IP**](https://setakit.com/wp-admin/post.php?post=3759&action=edit) استفاده می کنند.

تفاوت های آن ها به شرح زیر است:

**۱- Connection and connection-less**

TCP یک پروتکل connection-oriented و UDP یک پروتکل connection-less است. TCP قبل از ارسال اطلاعات، ارتباطی بین فرستنده و گیرنده برقرار می کند در صورتی که UDP قبل از ارسال داده ها، اتصال برقرار نمی کند.

**۲- Reliability**

TCP قابل اعتماد است به طوری که تضمین می کند داده های ارسال شده را به گیرنده تحویل دهد. چنانچه قسمتی از اطلاعات در فرایند ارسال از بین برود، آن را بازیابی و مجددا ارسال می کند. TCP مدام در حال بررسی [**پکت داده ها**](https://setakit.com/packet/) جهت احتمال خطا می باشد و آن ها را ردیابی نموده تا اطلاعاتی از بین نرود و یا خراب نشود.

UDP قابل اعتماد نیست، تحویل داده های ارسال شده را توسط گیرنده، تضمین نمی کند و حتی اطلاعات ارسال شده، ممکن است از بین برود و یا خراب شود.

**۳- Flow control**

TCP از مکانیزم کنترل جریان استفاده می کند تا مطمئن شود، فرستنده به یکباره حجم زیادی از اطلاعات را ارسال نکند. عملکرد TCP برای حل این مسئله به این شکل است که مقداری از داده ها در یک Send Buffer و Receive Buffer نگهداری می کند و در زمان مناسب ارسال و یا دریافت می کند.

زمانی که برنامه، آماده دریافت داده می باشد، اطلاعات را از Receive Buffer می خواند و دریافت می کند، چنانچه Receive Buffer، پر باشد، دریافت کننده قادر به دریافت اطلاعات نمی باشد پس منجر به از دست رفتن اطلاعات می شود. بنابراین جهت حفظ و نگهداری داده ها، دریافت کننده میزان فضای خالی Receive Buffer را برای فرستنده مشخص می کند. گیرنده، با هر بار دریافت packet، میزان فضای خالی را برای فرستنده مشخص می کند.

UDP از جریان کنترل استفاده نمی کند؛ بنابراین، packet ها در فرایند جریان مداوم، یا ارسال می شوند و یا از بین می روند.

**۴- Ordering**

TCP تضمین می کند پکت ها، به ترتیب و بر اساس تاریخ ارسال شوند، در صورتی که ترتیب ارسال UDP بر هیچ مبنای خاصی قرار ندارد.

**۵- Speed**

سرعت TCP، به این دلیل که عملکرد گسترده ای دارد، خیلی کندتر از UDP است. TCP می بایست یک connection ایجاد کند، خطاهای احتمالی را بررسی کند و همچنین فایل ها را، همان طور که ارسال شده، دریافت کند.

**۶- Usage**

از TCP برای برنامه هایی استفاده می شود که نیاز به قابلیت اطمینان بالایی وجود داشته باشد و سرعت در آن ها خیلی اهمیت نداشته باشد.

* World Wide Web (HTTP, HTTPS)
* Secure Shell (SSH)
* File Transfer Protocol (FTP)
* Email ([**SMTP**](https://setakit.com/%d9%8fsmtp/), [**IMAP/POP**](https://setakit.com/imap-vs-pop3/))

از UDP برای برنامه هایی استفاده می شود که نیاز به سرعت و کارایی داشته باشند.

* Streaming videos
* Online games
* Live broadcasts
* Domain Name System ([**DNS**](https://setakit.com/dns/))
* Voice over Internet Protocol (VoIP)
* Trivial File Transfer Protocol (TFTP)