

UDP

UDP ou "User Datagram Protocol" est un protocole de communication IP sans établissement de session ou accusés de réception. Il permet le transfert de données à une très faible latence mais sans garantie de livraison et donc une moindre fiabilité que certains autres protocoles.

- [Cas d'utilisation](#)
- [Fonctionnement](#)

Cas d'utilisation

Applications sensibles à la latence

Certains mécanismes de communication réseau sont plus sensibles à la latence. On peut penser ici à des appels téléphoniques ou des jeux vidéos en ligne. L'utilisation de UDP ne nécessitant pas d'établissement de session avant un transfert de données permet donc l'établissement d'une connexion et le transfert d'une donnée avec un délai de livraison minimal.

Application en direct

Si une connexion est un flux en direct, il est parfois préférable d'avoir une latence réduite en prenant le risque qu'une donnée ne soit pas livrée. On peut ici penser à des caméras de surveillance ou de téléopération. L'utilisation de UDP permet de minimiser la latence et si une image du flux vidéo est perdue en chemin, l'image suivante mettra à jour l'affichage de toute façon.

Dans la transmission d'un flux vidéo, seulement les modifications sont transférées d'une image à une autre mais suite à un certain nombre de modifications d'image, une image clé contenant l'ensemble de l'image est transférée. Lorsque cette image clé est perdue en chemin, il y a de fortes chances que l'image soit distordue comme ceci :



Dans le contexte d'un jeu vidéo, une donnée qui serait perdue/abandonnée entre deux joueurs (si le jeu est en pair-à-pair) ou entre un joueur et le serveur pourrait avoir comme impact que lorsque la donnée suivante soit acheminée, le joueur risque de "rubber band" (retourner à l'arrière) puisque les autres joueurs ou le serveur n'auront pas été informés des derniers mouvements du joueur.



Applications de "broadcast"

Puisqu'un broadcast n'est pas destiné à une seule adresse IP, il n'est généralement pas possible d'établir une session entre deux appareils. Il est seulement possible de diffuser une donnée et laisser le domaine de diffusion acheminer la donnée. Certaines applications telles que [DHCP](#) ne nécessitent pas d'adresse IP pour fonctionner, ceux-ci sont généralement des protocoles UDP.

Petits transferts de données

On peut penser ici à une requête [DNS](#) ou une synchronisation d'horloge réseau par [NTP](#). Ces transferts de données sont minimes et en cas de donnée perdue, effectuer une nouvelle requête devrait prendre un temps négligeable.

Fonctionnement

Chaque protocole IP a un numéro lui étant associé, dans le cas de UDP, il s'agit du protocole IP #17.

UDP est un des protocoles IP les plus simples dans son fonctionnement. L'en-tête d'un paquet UDP contient généralement seulement le port de source (16 bits), de destination (16 bits), la longueur de la donnée livrée (16 bits) et une donnée permettant de valider la donnée livrée à l'intérieur du paquet (aussi 16 bits). Cette en-tête a donc une longueur de seulement 64 bits.

Puisque UDP ne garanti pas la livraison d'une donnée, il n'est pas non plus garanti que les données seront livrées dans le bon ordre. Si une donnée doit être segmentée et réassemblée du côté du client, c'est au niveau de la couche d'application que les données devront être étiquetées pour être interprétées dans l'ordre par le client. La somme de contrôle permettant de confirmer la validité de la donnée livrée est le seul mécanisme pour s'assurer que la donnée acheminée ne soit pas erronée.

Dans l'exemple d'en-tête suivante, une requête UDP est effectuée vers le port UDP 443 (QUIC) d'un serveur web et le message aura une longueur de 31 octets.

```
User Datagram Protocol, Src Port: 63857, Dst Port: 443
  Source Port: 63857
  Destination Port: 443
  Length: 39
  Checksum: 0x875d [unverified]
  UDP payload (31 bytes)
```

Le serveur nous répond ensuite au port source de notre requête initiale avec un message composé dans ce cas-ci de 24 octets.

```
User Datagram Protocol, Src Port: 443, Dst Port: 63857
  Source Port: 443
  Destination Port: 63857
  Length: 32
  Checksum: 0x46a0 [unverified]
  UDP payload (24 bytes)
```