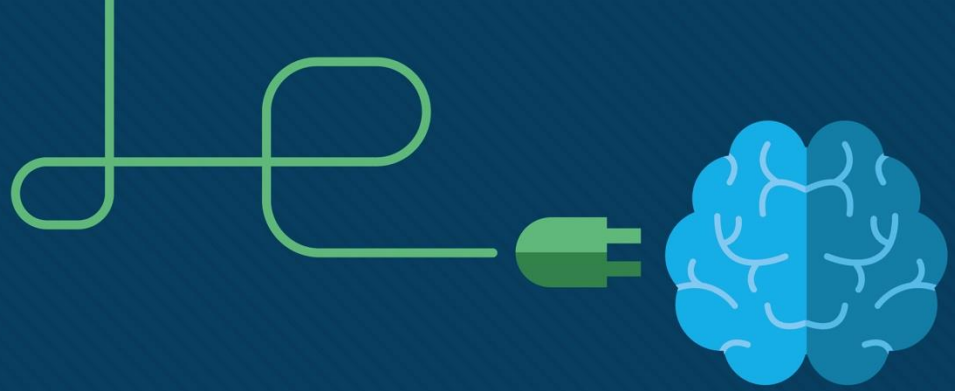


Module 14: La couche de transport

Contenu Pédagogique de l'instructeur

Présentation des réseaux V7.0
(ITN)

Objectifs du module

Titre du module: Couche de transport

L'objectif du module: Comparer les opérations des protocoles de la La couche de transport dans la prise en charge de la communication de bout en bout.

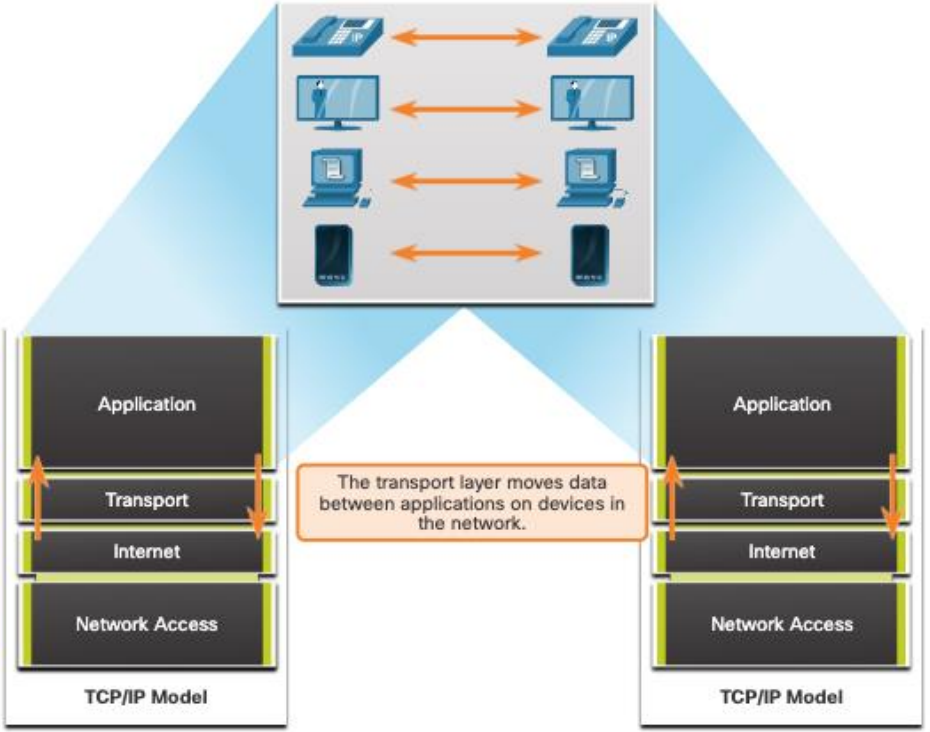
Titre du rubrique	Objectif du rubrique
Transport des données	Expliquer le rôle de la couche de transport dans la gestion du transport des données dans une communication de bout en bout.
Présentation du protocole TCP	Expliquer les caractéristiques du TCP.
Présentation du protocole UDP	Expliquer les caractéristiques de l'UDP.
Numéros de port	Expliquer comment TCP et UDP utilisent les numéros de port.
Processus de communication TCP	Expliquer comment les processus d'établissement et d'interruption de session TCP garantissent la fiabilité des communications.
Fiabilité et contrôle des flux	Expliquer comment les unités de données de protocole TCP sont transmises et comment leur réception est confirmée pour garantir l'acheminement des données.
Communication UDP	Comparer les opérations des protocoles de la La couche de transport dans la prise en charge de la communication de bout en bout.

14.1 Transport des données

Rôle de la couche transport

La couche de transport est:

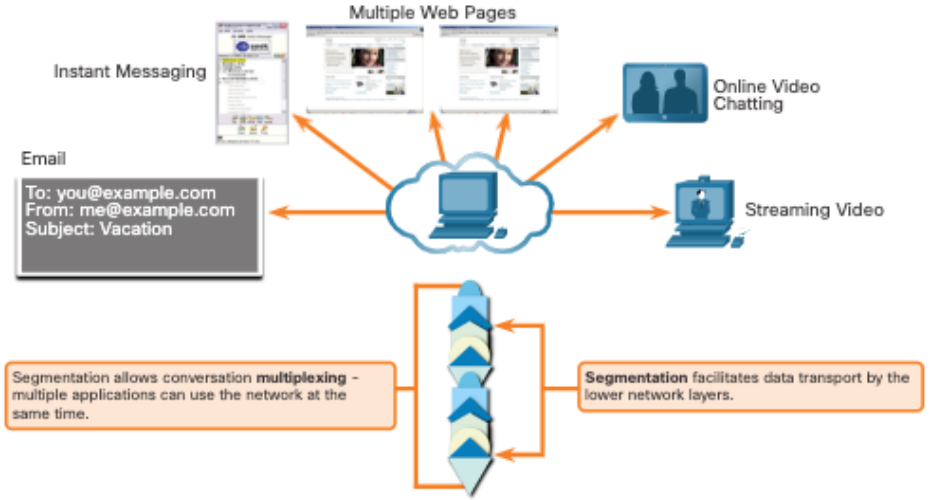
- responsable des communications logiques entre les applications exécutées sur différents hôtes.
- La liaison entre la couche d'application et les couches inférieures qui sont responsables de la transmission du réseau.



Responsabilités de la La couche de transport

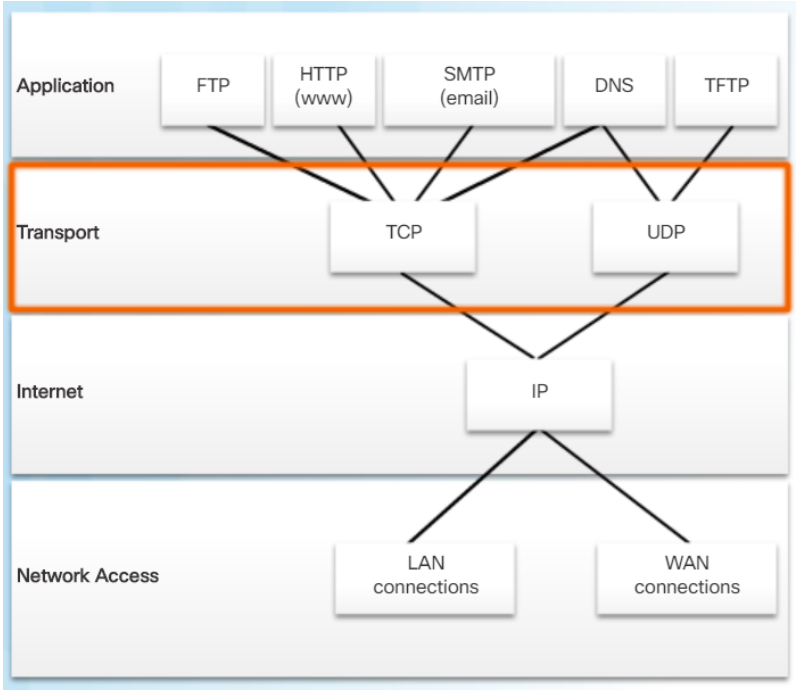
La couche de transport a les responsabilités suivantes:

- Suivre les conversations individuelles
- Segmentation des données et reconstitution des segments
- Ajouter les informations d'en-tête
- Identifier, séparer et gérer plusieurs conversations
- Utiliser la segmentation et le multiplexage pour permettre à différentes conversations de communication d'être entrelacées sur le même réseau



Protocoles de la couche de transport

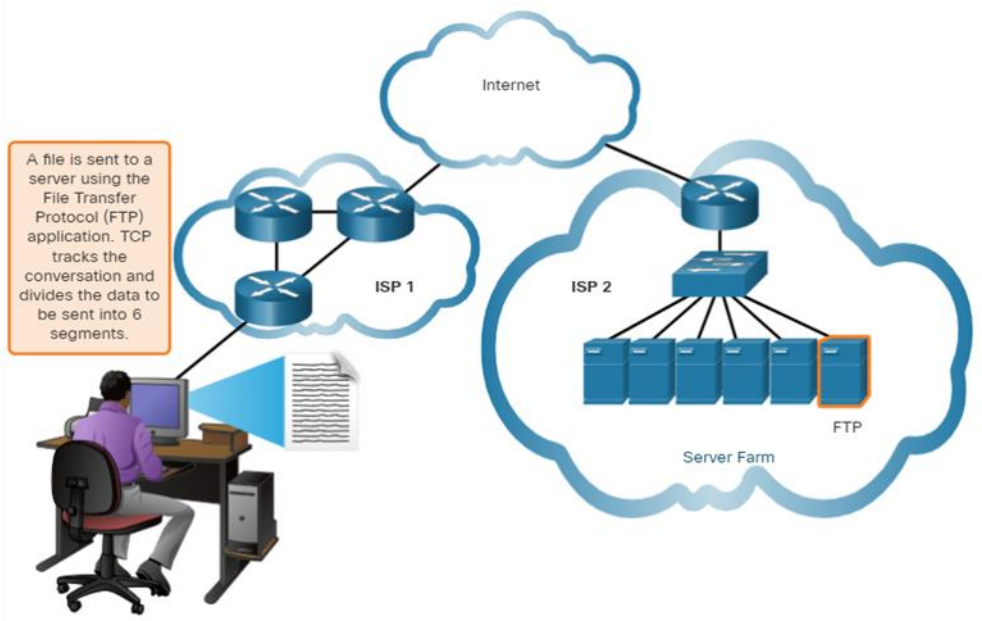
- Il ne fixe pas le mode d'acheminement ou de transport des paquets.
- Les protocoles de La couche de transport spécifient comment transférer des messages entre les hôtes et ils sont responsables de la gestion des exigences de fiabilité d'une conversation.
- La couche transport comprend les protocoles TCP et UDP.



Protocole TCP (Transmission Control Protocol)

TCP assure la fiabilité et le contrôle du flux. Les opérations de base de TCP:

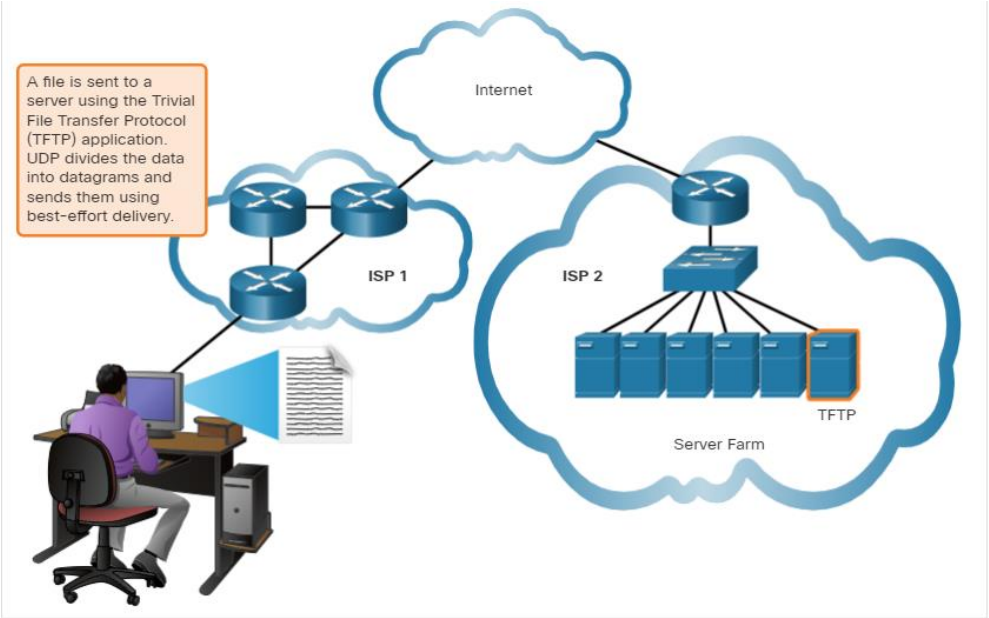
- Numéroter et suivre les segments de données transmis à un hôte spécifique à partir d'une application spécifique
- Accuser la réception des données reçues
- Retransmettre toute donnée non reconnue après un certain temps
- Séquence des données qui pourraient arriver dans un ordre incorrect
- Envoyer des données à un taux efficace et acceptable par le



UDP (Transport of Data User Datagram Protocol)

UDP fournit des fonctions de base permettant d'acheminer des segments de données entre les applications appropriées tout en ne nécessitant que très peu de surcharge et de vérification des données.

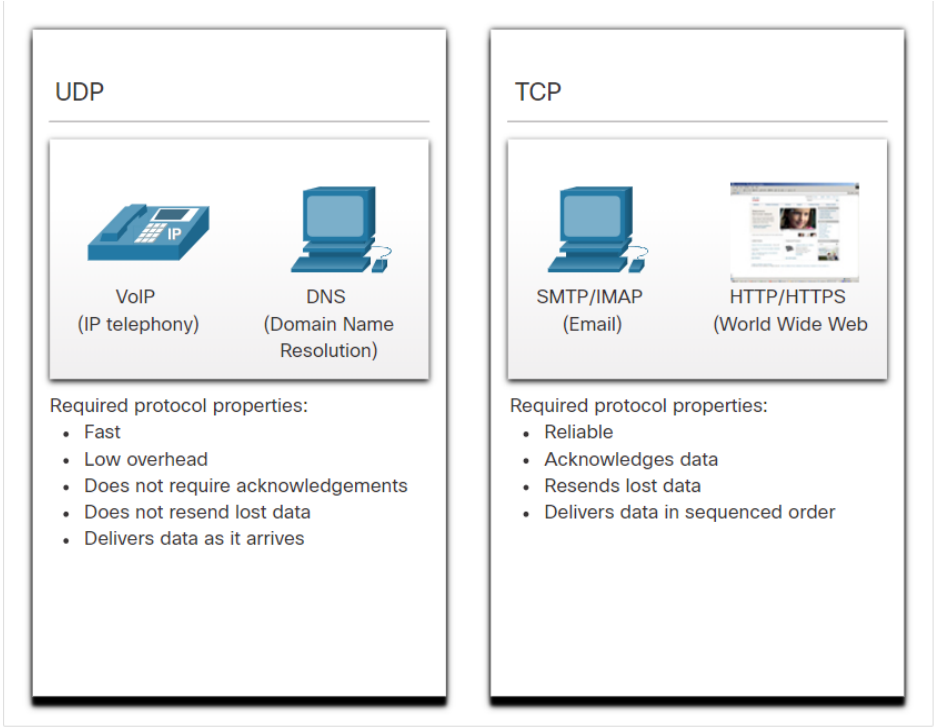
- UDP est un protocole sans connexion.
- UDP est également connu comme un protocole de livraison du meilleur effort, car il n'y a pas d'accusé de réception des données à la destination.



La choix du protocole de couche transport le mieux adapté à une application donnée

UDP est également utilisé par les applications de requête et de réponse où les données sont minimales, et la retransmission peut être effectuée rapidement.

S'il est important que toutes les données arrivent et qu'elles puissent être traitées dans leur ordre approprié, TCP est utilisé comme protocole de transport.



14.2 Présentation du protocole TCP

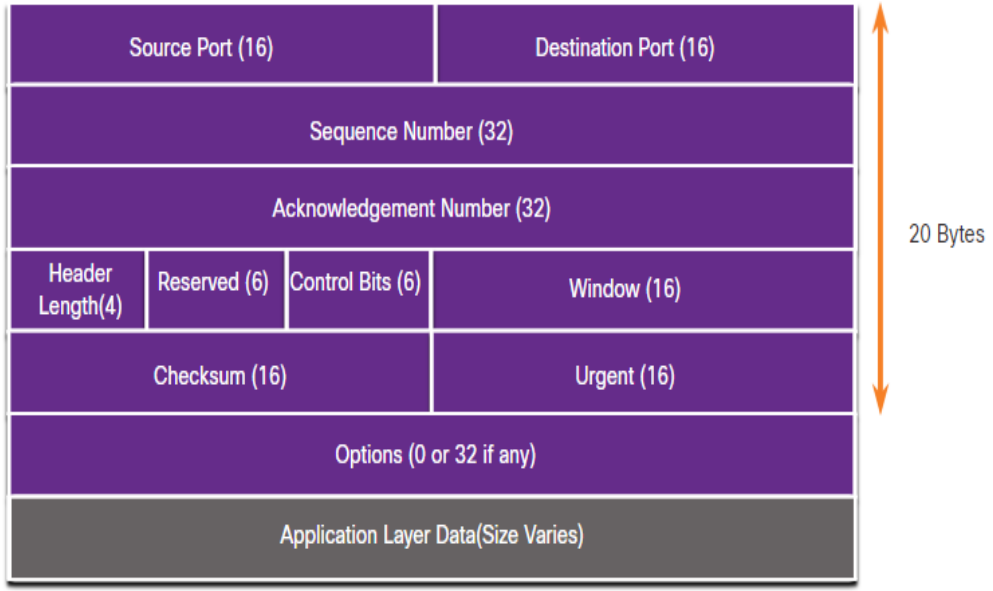
Caractéristiques du protocole TCP

- **Établit une session** -TCP est un protocole connexion orienté qui négocie et établit une connexion (ou session) permanente entre les appareils source et destination avant le transmettre du trafic.
- **Garantit une livraison fiable** - Pour de nombreuses raisons, il est possible qu'un segment soit corrompu ou complètement perdu lors de sa transmission sur le réseau. TCP s'assure que chaque segment envoyé par la source arrive à la destination
- **Fournit la livraison dans le même ordre** - Étant donné que les réseaux peuvent fournir plusieurs routes dont les débits de transmission varient, les données peuvent arriver dans le mauvais ordre.
- **Soutien le contrôle de flux** - Les hôtes du réseau ont des ressources limitées (par exemple la mémoire et la puissance de traitement). Quand le protocole TCP détermine que ces ressources sont surexploitées, il peut demander à l'application qui envoie les données d'en réduire le flux.

L'en-tête TCP

TCP est un protocole avec état, ce qui signifie qu'il garde une trace de l'état de la session de communication.

le protocole TCP enregistre les informations qu'il a envoyées et les informations qu'il a reçues.

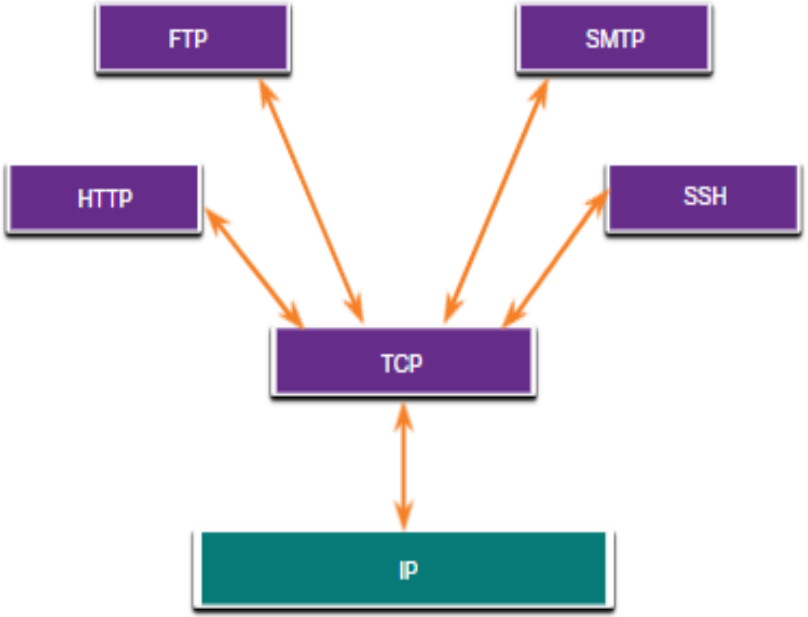


TCP Champs d'en-tête

Champ d'en-tête TCP	Description
Port source	Champ 16 bits utilisé pour identifier l'application source par le numéro de port.
Port de destination	Champ de 16 bits utilisé pour identifier l'application de destination par le numéro de port.
Numéro de séquence	Champ 32 bits utilisé à des fins de réassemblage de données.
Numéro d'accusé de réception	Champ 32 bits est utilisé pour indiquer que les données ont été reçues et l'octet suivant est prévu de la source.
Longueur d'en-tête	Champ 4 bits connu sous le nom de « offset de données » qui indique la longueur de l'en-tête du segment TCP.
Réservé	Un champ de 6 bits qui est réservé pour une utilisation future.
Bits de contrôle	Un champ de 6 bit utilisé comprennent des codes de bits qui indiquent l'objectif et la fonction du segment TCP.
Taille de fenêtre	Champ 16 bits utilisé pour indiquer le nombre d'octets qui peut être acceptés.
Somme de contrôle	Un champ de 16 bits utilisé pour la vérification des erreurs de l'en-tête du segment et des données.
Urgent	Champ 16 bits utilisé pour indiquer si les données contenues sont urgentes.

Applications utilisant le protocole TCP

Le TCP gère toutes les tâches associées à la division du flux de données en segments, à la fiabilité, au contrôle du flux de données et à la réorganisation des segments.



14.3 Présentation du protocole UDP

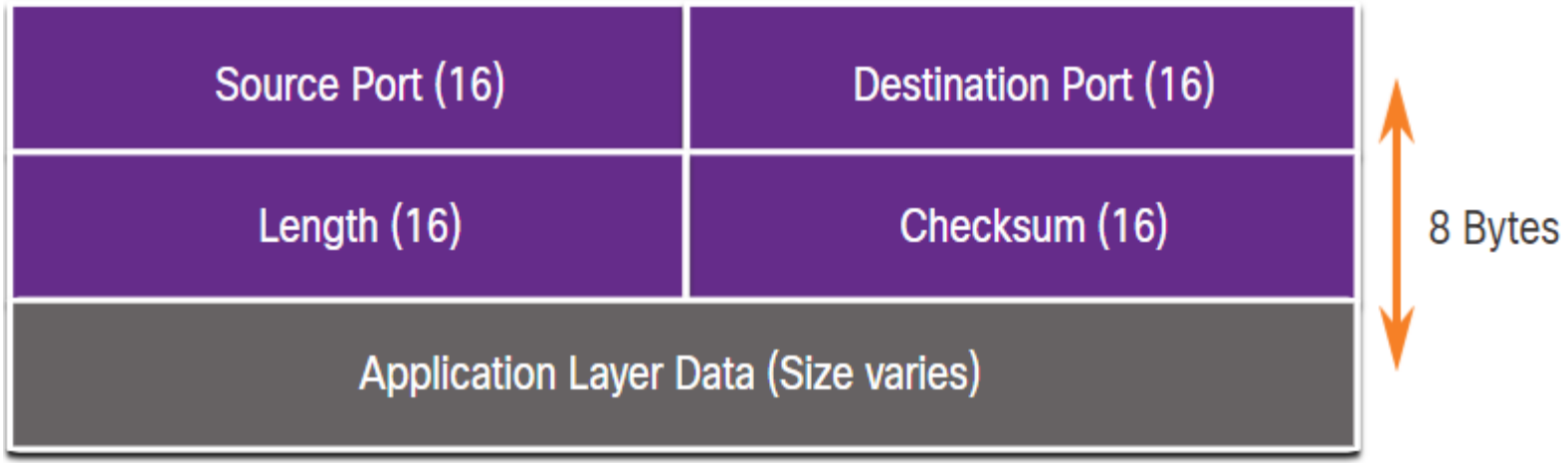
Caractéristiques du protocole UDP

Les caractéristiques de l'UDP sont les suivantes :

- Les données sont reconstituées selon l'ordre de réception.
- Les segments qui sont perdus ne sont pas renvoyés.
- Il n'y a pas d'établissement de session.
- L'expéditeur n'est pas informé de la disponibilité des ressources.

L'en-tête UDP

L'en-tête UDP est beaucoup plus simple que l'en-tête TCP car il n'a que quatre champs et nécessite 8 octets (c'est-à-dire 64 bits).



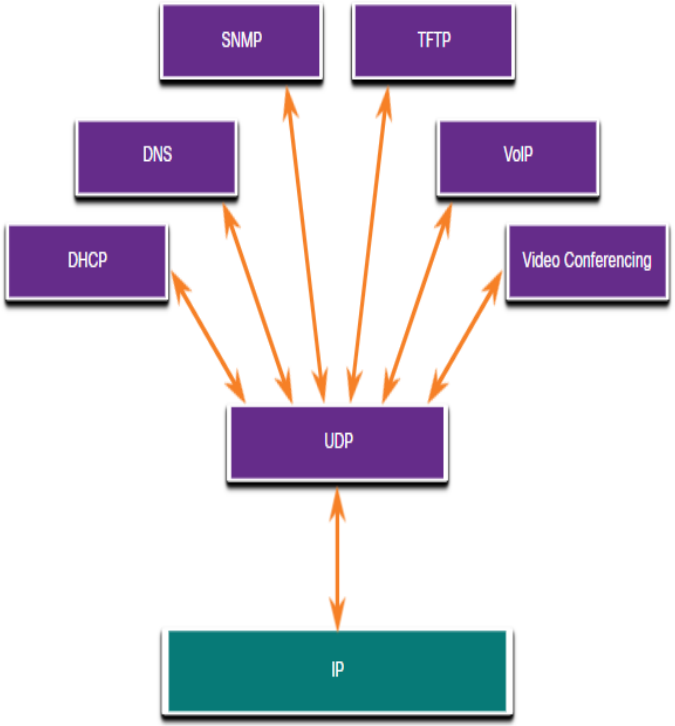
Les Champs d'en-tête UDP

Le tableau identifie et décrit les quatre champs d'un en-tête UDP.

Champ d'en-tête UDP	Description
Port source	Champ 16 bits utilisé pour identifier l'application source par le numéro de port.
Port de destination	Champ de 16 bits utilisé pour identifier l'application de destination par le numéro de port.
Longueur	Champ 16 bits indiquant la longueur de l'en-tête de datagramme UDP.
Somme de contrôle	Champ 16 bits utilisé pour la vérification des erreurs de l'en-tête et des données du datagramme.

Applications utilisant le protocole UDP

- Les applications vidéo et multimédia en direct : Ces applications peuvent tolérer une certaine perte de données, mais ne nécessitent que peu ou pas de délai. La voix sur IP et le streaming vidéo sont de bons exemples.
- Les simples applications de requête et de réponse : ils sont des applications dont les transactions sont simples et pour lesquelles un hôte envoie une requête à laquelle il recevra ou non une réponse. Exemples incluent DNS et DHCP.
- Applications qui gèrent elles-mêmes la fiabilité- Communications unidirectionnelles où le contrôle de flux, la détection des erreurs, les accusés de réception et la récupération des erreurs ne sont pas nécessaires, ou peuvent être gérés par



14.4 Les Numéros de ports

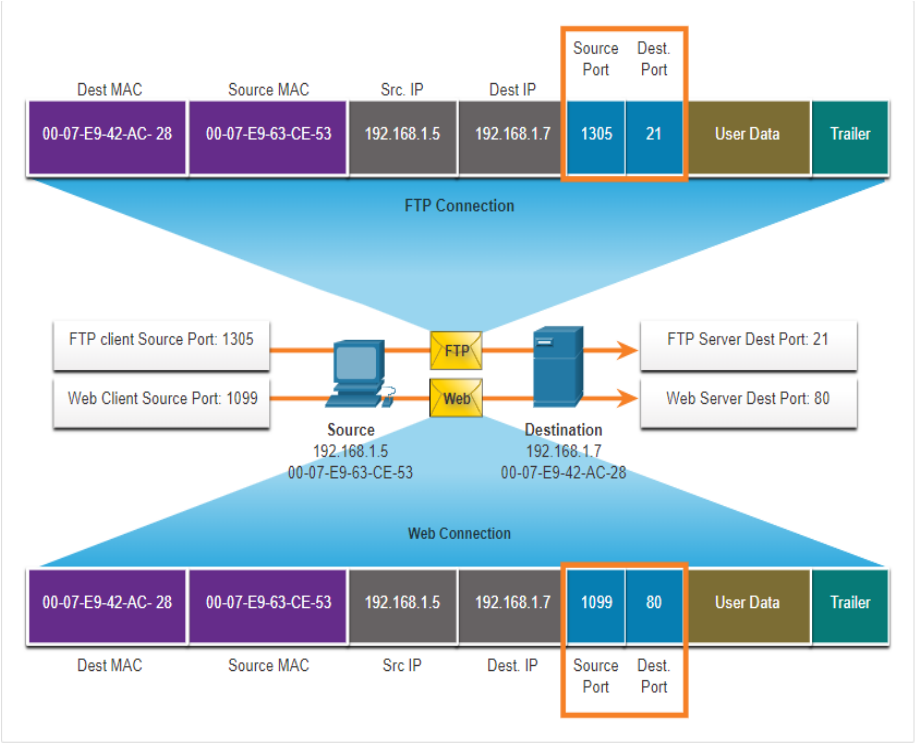
Communications multiples et séparées

Les protocoles de couches de transport TCP et UDP utilisent des numéros de port pour gérer plusieurs conversations simultanées.

Le numéro de port source est associé à l'application d'origine sur l'hôte local tandis que le numéro de port de destination est associé à l'application de destination sur l'hôte distant.

Paires d'interfaces de connexion

- Les ports sources et de destination sont placés à l'intérieur du segment.
- Les segments sont ensuite encapsulés dans un paquet IP.
- La combinaison de l'adresse IP source et du numéro de port source, ou de l'adresse IP de destination et du numéro de port de destination, est appelée interface de connexion.
- Les interfaces de connexion permettent à plusieurs processus exécutés sur un client de se différencier les uns des autres, et aux multiples connexions à un processus serveur de se distinguer les unes des autres.



Groupes de numéros de port

Groupe de ports	Gamme de numéros	Description
Ports réservés	**de 0 à 1023**	•Ces numéros de port sont réservés aux services et aux applications courants ou populaires tels que les navigateurs web, les clients de messagerie électronique et les clients d'accès à distance. •Les ports bien connus définis pour les applications serveur courantes permettent aux clients d'identifier facilement le service associé requis.
Ports inscrits	**de 1024 à 49151**	•ces numéros de port sont affectés par l'IANA à une entité demandeuse pour être utilisés avec des processus ou des applications spécifiques. •Ces processus sont essentiellement des applications particulières qu'un utilisateur a choisi d'installer plutôt que des applications courantes qui recevraient un numéro de port réservé. •Par exemple, Cisco a enregistré le port 1812 pour son processus d'authentification du serveur RADIUS.
Ports privés et/ou **dynamiques**	**de 49152 à 65535**	•Ces ports sont également connus sous le nom de *ports éphémères*. •Le système d'exploitation du client attribue généralement des numéros de port dynamique lorsqu'une connexion à un service est lancée. •Le port dynamique est ensuite utilisé pour identifier l'application cliente pendant la communication.

Groupes de numéros de port (Suite)

Numéros de ports reconnus

Numéro de port	Protocole	Application
20	TCP	FTP (File Transfer Protocol) - Données
21	TCP	File Transfer Protocol (FTP) - Contrôle
22	TCP	SSH (Secure Shell)
23	TCP	Telnet
25	TCP	Protocole SMTP
53	UDP, TCP	Service de noms de domaine (Domain Name Service, DNS)
67	UDP	Serveur DHCP (Dynamic Host Configuration Protocol)
68	UDP	Client DHCP (Dynamic Host Configuration Protocol)
69	UDP	Protocole TFTP (Trivial File Transfer Protocol)
80	TCP	Protocole HTTP (Hypertext Transfer Protocol)
110	TCP	Protocole POP3 (Post Office Protocol version 3)
143	TCP	IMAP (Internet Message Access Protocol)
161	UDP	Protocole SNMP (Simple Network Management Protocol)
443	TCP	protocole HTTPS (Hypertext Transfer Protocol Secure)

La commande netstat

Les connexions TCP inexpliquées peuvent poser un risque de sécurité majeur. Netstat est un outil important pour vérifier les connexions.

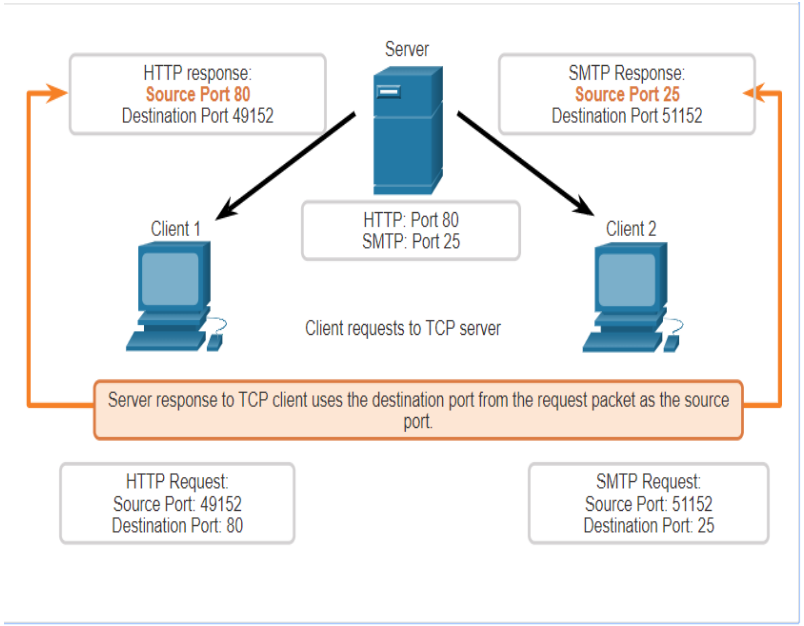
```
C:\> netstat
Active Connections
Proto Local Address Foreign Address State
TCP 192.168.1. 124:3126 192.168.0.2:netbios-ssn ESTABLISHED
TCP 192.168.1. 124:3158 207.138.126.152:http ESTABLISHED
TCP 192.168.1.124:3159 207.138.126.169:http ESTABLISHED
TCP 192.168.1.124:3160 207.138.126.169:http ESTABLISHED
TCP 192.168.1. 124:3161 sc.msn.com:http ESTABLISHED
TCP 192.168.1.124:3166 www.cisco.com:http ESTABLISHED
```

14.5 Processus de communication TCP

Processus de serveur TCP

Chaque processus de demande s'exécutant sur un serveur est configuré pour utiliser un numéro de port.

- Deux services ne peuvent pas être affectés au même numéro de port d'un serveur au sein des mêmes services de la couche transport.
- Une application de serveur active affectée à un port spécifique est considérée comme étant ouverte, ce qui signifie que la couche transport accepte et traite les segments adressés à ce port.
- Toute demande entrante d'un client qui est adressée à l'interface de connexion correcte est acceptée et les données sont transmises à l'application de serveur.

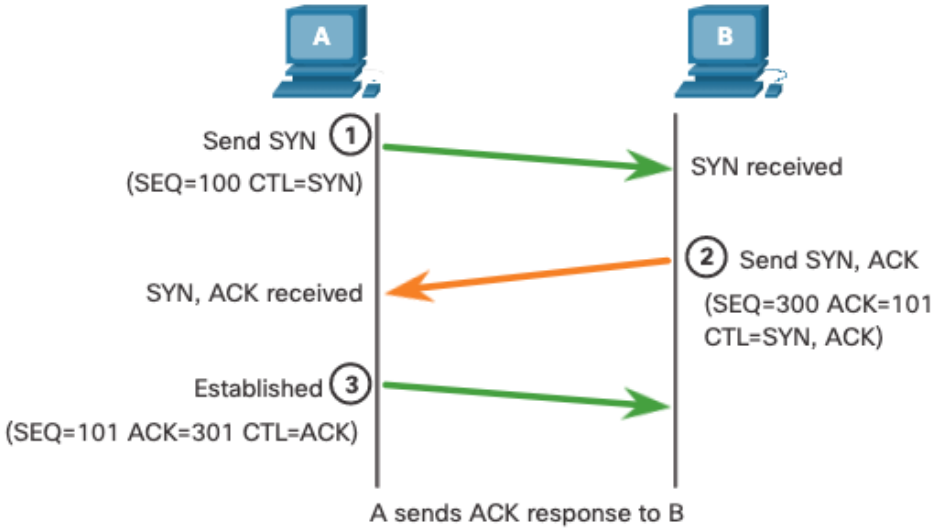


Établissement d'une connexion TCP

Étape 1: Le client demande l'établissement d'une session de communication client-serveur avec le serveur.

Étape 2: Le serveur accuse la réception de la session de communication client-serveur et demande l'établissement d'une session de communication serveur-client.

Étape 3: Le client accuse réception de la session de communication serveur-client.



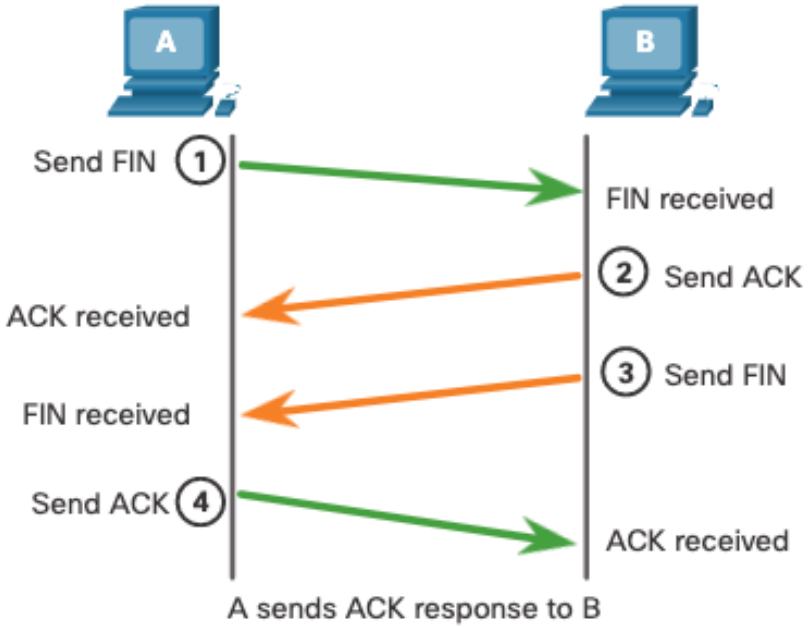
La fin de session TCP

Étape 1: Quand le client n'a plus de données à envoyer dans le flux, il envoie un segment dont l'indicateur FIN est défini.

Étape 2:Le serveur envoie un segment ACK pour indiquer la bonne réception du segment FIN afin de terminer la session du client au serveur.

Étape 3:Le serveur envoie un segment FIN au client pour terminer la session du serveur au client.

Étape 4:Le client répond à l'aide d'un segment ACK pour accuser la réception du segment FIN envoyé par le serveur.



Analyse de la connexion TCP en trois étapes

La connexion en trois étapes:

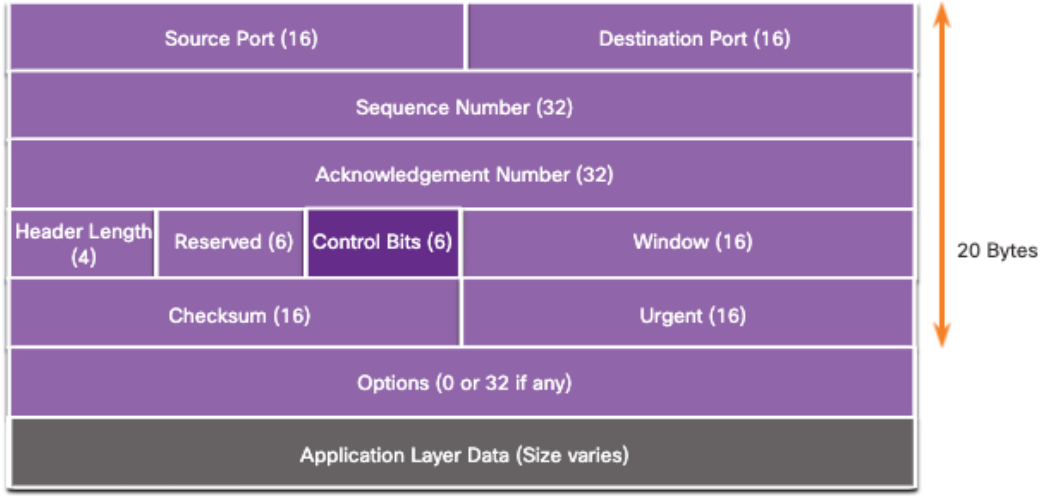
- Elle vérifie que le périphérique de destination est bien présent sur le réseau.
- Elle s'assure que le périphérique de destination a un service actif et qu'il accepte les requêtes sur le numéro de port de destination que le client qui démarre la session a l'intention d'utiliser.
- Elle informe le périphérique de destination que le client source souhaite établir une session de communication sur ce numéro de port.

Une fois la communication est terminée, les sessions sont terminées et la connexion est interrompue. Les mécanismes de connexion et de session permettent la fonction de fiabilité du TCP.

Analyse de la connexion TCP en trois étapes (Suite)

Les six indicateurs de bits de contrôle sont les suivants:

- **URG** - Champ de pointeur urgent significatif (Urgent pointer field significant)
- **ACK** - Indicateur d'accusé de réception utilisé dans l'établissement de la connexion et la fin de la session
- **PSH** - Fonction push (Push function)
- **RST** - Réinitialisation de la connexion en cas d'erreur ou de dépassement de délai
- **SYN** - Synchroniser les numéros de séquence utilisés dans l'établissement de connexion
- **FIN** - Plus de données de l'expéditeur et utilisées dans la fin de session



Démonstration vidéo – Connexion TCP en trois étapes

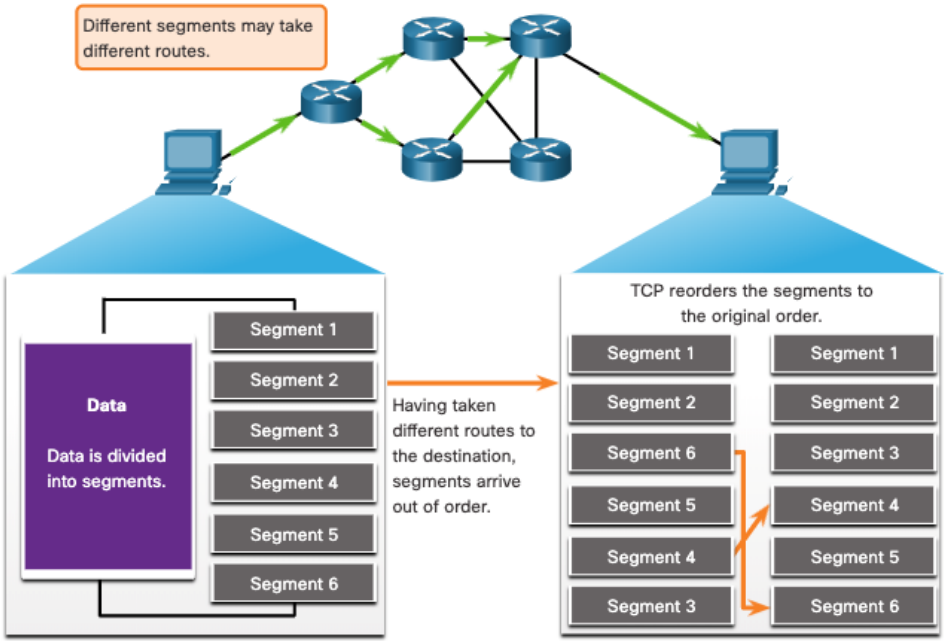
Cette vidéo présentera les points suivants:

- La connexion TCP en trois étapes
- Terminaison d'une conversation TCP.

14.6 Fiabilité et contrôle de flux

Fiabilité du TCP - Livraison garantie et commandée

- TCP peut également aider à maintenir le flux des paquets afin que les périphériques ne soient pas surchargés.
- Il peut arriver que des segments TCP n'arrivent pas à la destination ou qu'ils soient hors d'usage.
- Toutes les données doivent être reçues et les données de ces segments doivent être réassemblées dans l'ordre d'origine.
- Pour cela, des numéros d'ordre sont affectés à l'en-tête de chaque paquet.



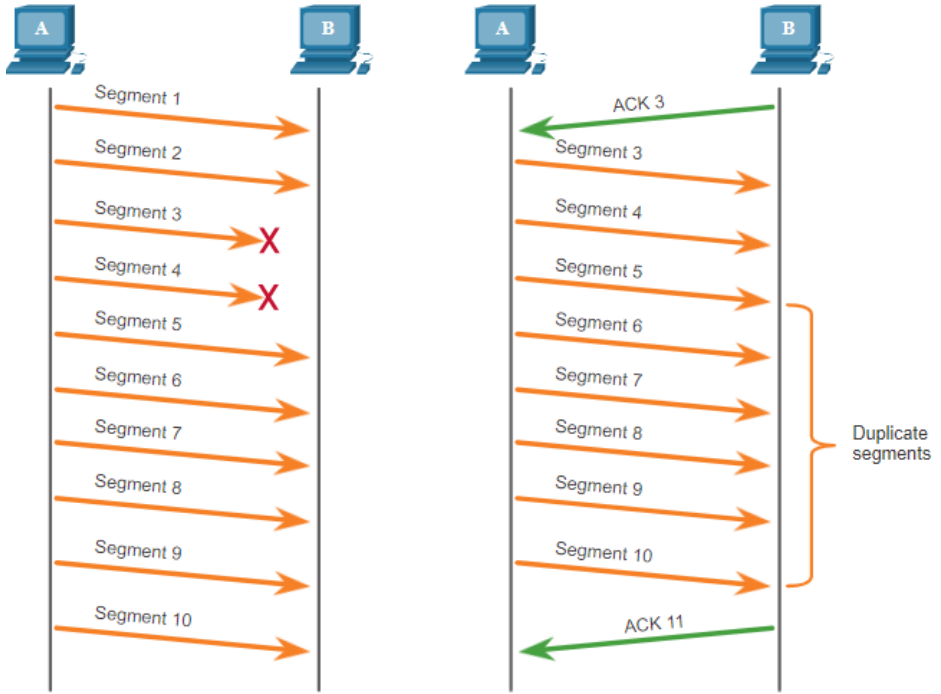
Démonstration vidéo -Fiabilité du protocole TCP - Numéros d'ordre et accusés de réception

Cette vidéo indique un exemple simplifié de fonctionnement du protocole TCP.

Fiabilité du TCP - Perte de données et retransmission

Quelle que soit la conception d'un réseau, la perte de données qui se produit occasionnellement.

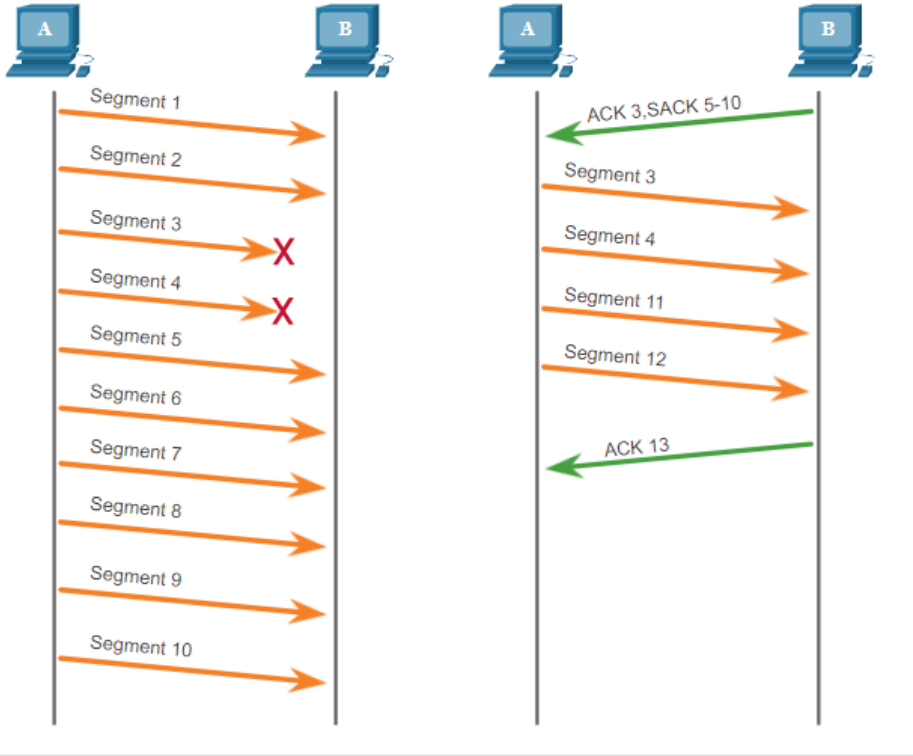
Le protocole TCP fournit des méthodes de gestion des pertes de segments. Parmi elles se trouve un mécanisme de retransmission des segments pour les données sans accusé de réception.



Fiabilité TCP — Perte et retransmission de données (Suite)

Aujourd'hui, les systèmes d'exploitation hôtes utilisent généralement une fonctionnalité TCP facultative appelée reconnaissance sélective (SACK), négociée au cours de la poignée de main à trois voies.

Si les deux hôtes prennent en charge SACK, le récepteur peut explicitement reconnaître quels segments (octets) ont été reçus, y compris les segments discontinus.



Démonstration vidéo – Perte de données et retransmission

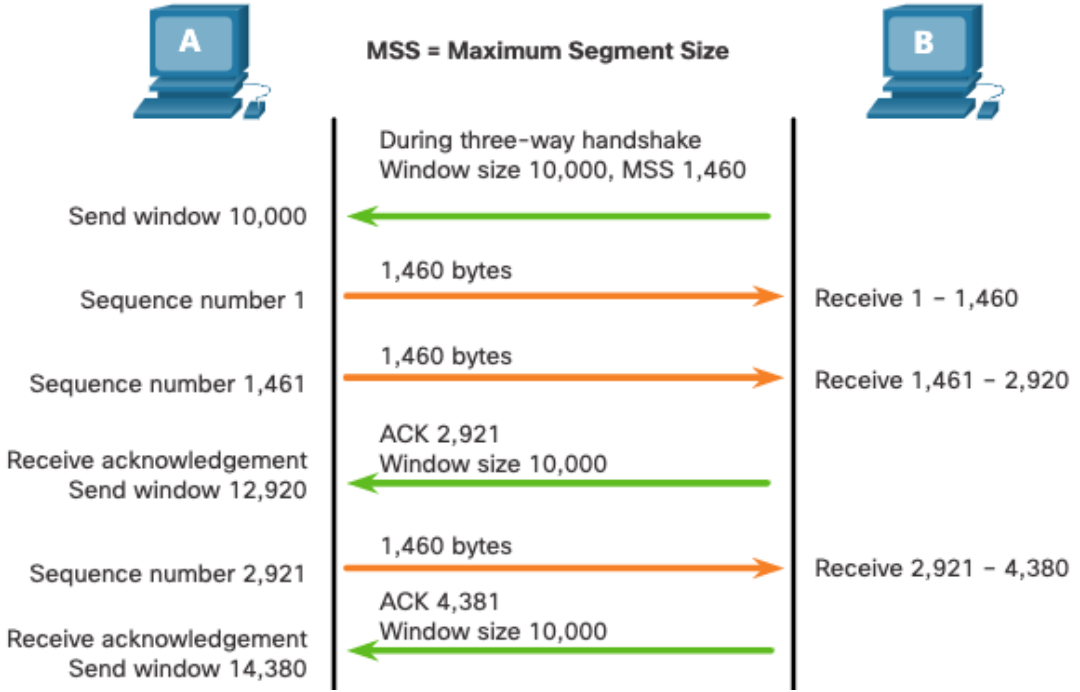
Cette vidéo indique le processus de renvoi des segments qui ne sont pas reçus initialement par la destination.

Contrôle de flux TCP – Taille de fenêtre et accusés de réception

Le protocole TCP offre des mécanismes de contrôle des flux comme suit:

- Le protocole TCP inclut également des mécanismes de contrôle de flux, qui correspond au volume de données que l'hôte de destination peut recevoir et traiter de manière fiable.

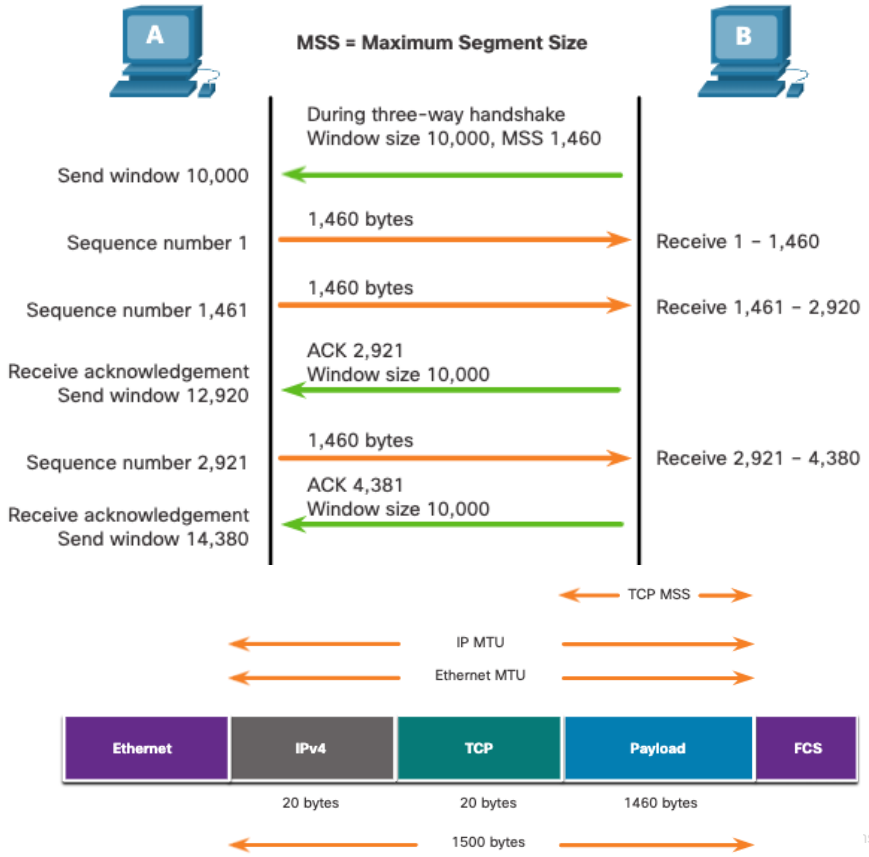
- Le contrôle de flux aide à maintenir la fiabilité des transmissions TCP en réglant le flux de données entre la source et la destination pour une session donnée.



Contrôle de flux TCP — Taille maximale du segment

La taille maximale du segment (MSS) est la quantité maximale de données que le périphérique de destination peut recevoir.

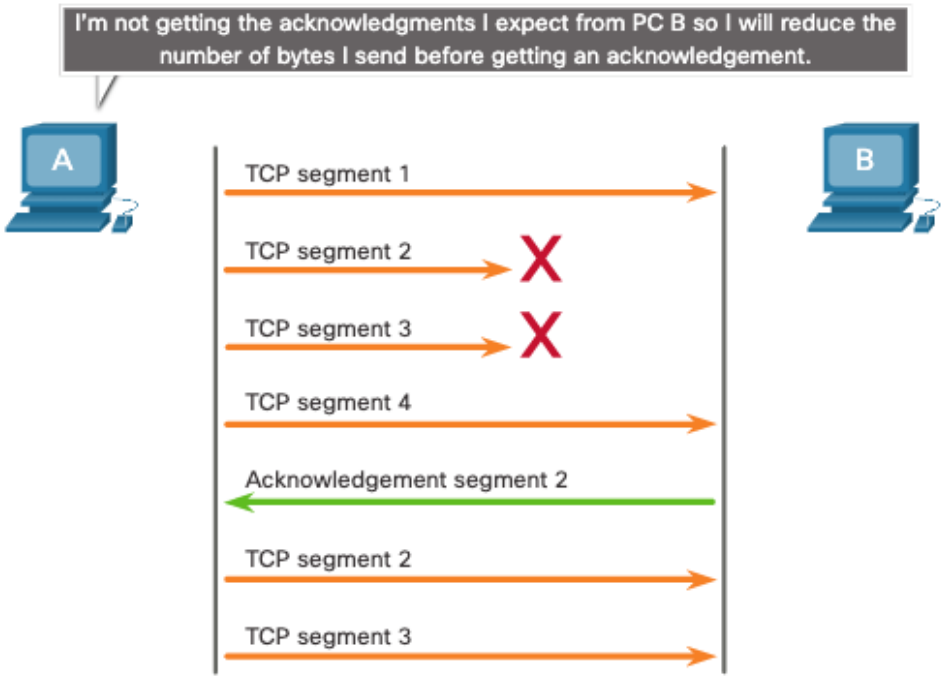
- Un MSS commun est de 1460 octets lors de l'utilisation d'IPv4.
- Un hôte détermine la valeur de son champ MSS en soustrayant les en-têtes IP et TCP de la MTU Ethernet, qui est 1500 octets par défaut.
- 1500 moins 60 (20 octets pour l'en-tête IPv4 et 20 octets pour l'en-tête TCP) laisse 1460 octets.



Contrôle de flux TCP – Prévention des encombrements

En cas d'encombrement sur un réseau, des paquets sont mis au rebut par le routeur surchargé.

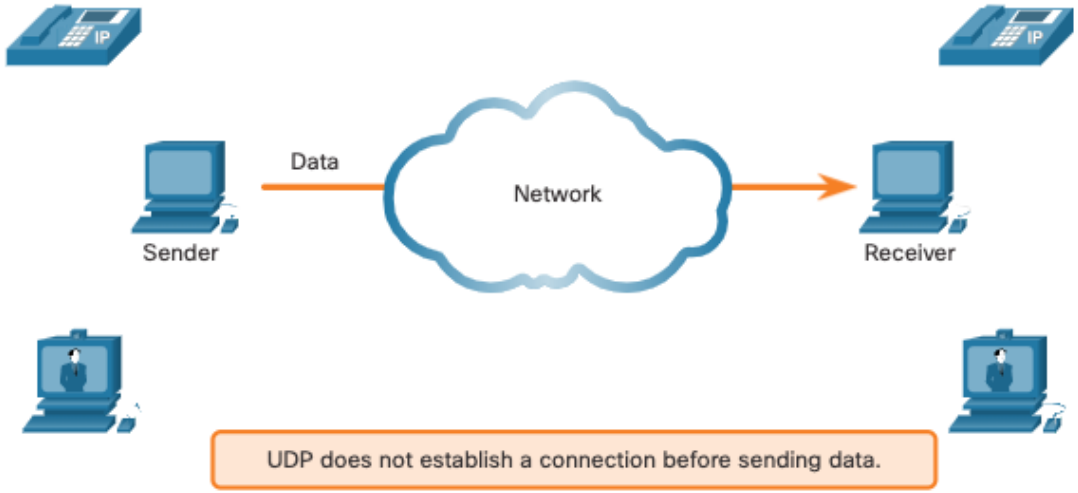
Afin d'éviter et de contrôler l'encombrement du réseau, le protocole TCP utilise divers mécanismes, minuteurs et algorithmes de gestion des encombrements.



14.7 Communication du protocole UDP

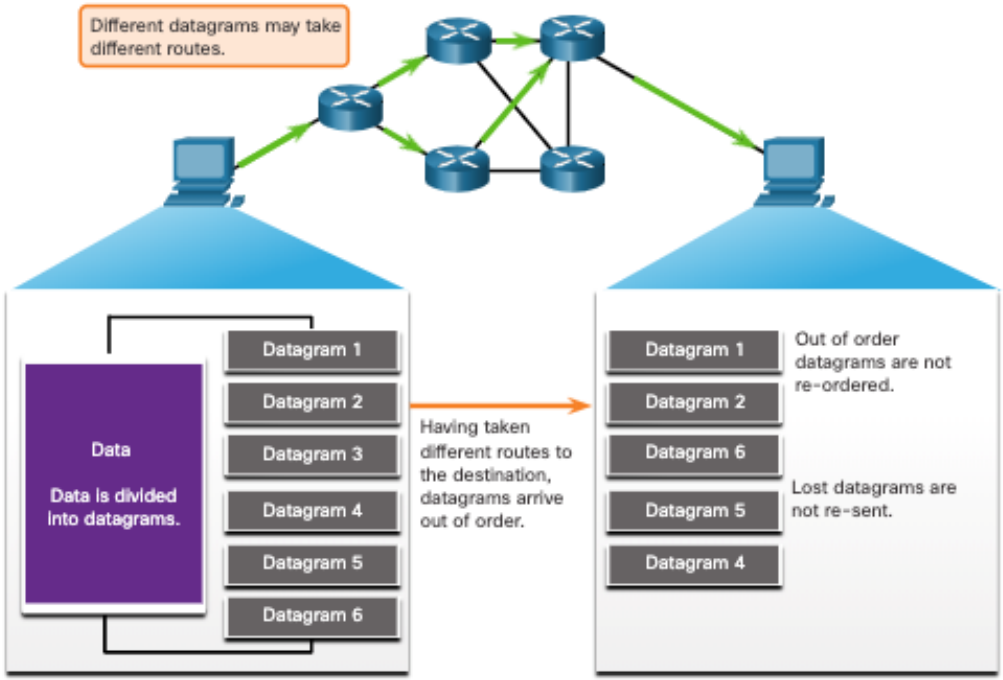
Faible surcharge et fiabilité du protocole UDP

Le protocole UDP n'établit pas de connexion. Le protocole UDP fournit un transport de données à faible surcharge car il utilise de petits en-têtes de datagrammes et n'offre pas de gestion du trafic réseau.



Réassemblage de datagrammes UDP

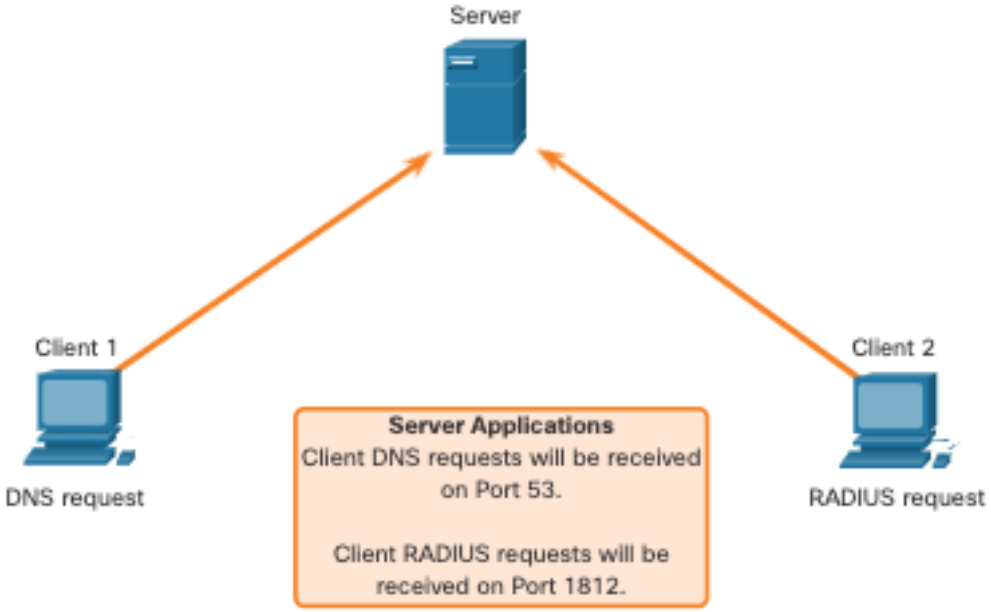
- Le protocole UDP n'effectue pas de suivi des numéros d'ordre comme le fait le protocole TCP.
- Il n'a en effet aucun moyen de réordonnancer les datagrammes pour leur faire retrouver leur ordre de transmission d'origine.
- Le protocole UDP se contente donc de réassembler les données dans l'ordre dans lequel elles ont été reçues, puis de les transmettre à l'application.



Processus et requêtes des serveurs UDP

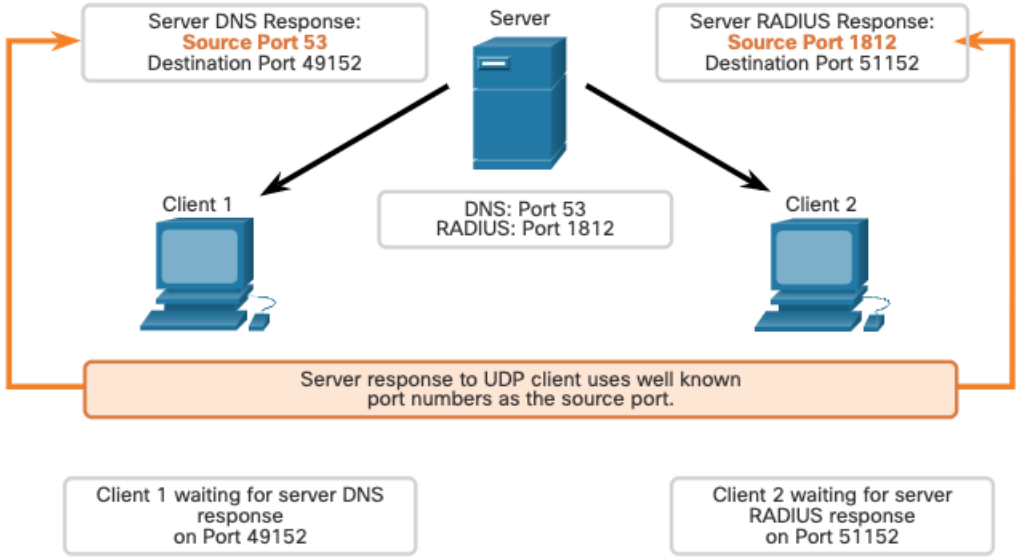
Les applications serveur basées sur l'UDP se voient attribuer des numéros de port connus ou enregistrés.

Le protocole UDP reçoit un datagramme destiné à l'un de ces ports, il transmet les données applicatives à l'application appropriée d'après son numéro de port.



Processus des clients UDP

- Le processus client UDP sélectionne dynamiquement un numéro de port dans une plage de numéros de ports et il l'utilise en tant que port source pour la conversation.
- Le port de destination est généralement le numéro de port réservé affecté au processus serveur.
- Une fois qu'un client a choisi le port source et le port de destination, la même paire de ports est utilisée dans l'en-tête de tous les datagrammes employés dans la transaction.




CISCO