

Client/serveur et protocoles applicatifs

Description du thème

Propriétés	Description
Intitulé long	Services et transport de paquets vers les applications : notions de client/serveur et protocoles applicatifs
Formation(s) concernée(s)	BTS Services Informatiques aux Organisations
Matière(s)	SI2
Présentation	<p>Les mécanismes abordés dans les séances précédentes (protocole Ethernet, résolution ARP, adressage IP, routage...) permettent à deux postes de communiquer sur un même réseau ou deux réseaux éloignés.</p> <p>Les applications utilisent l'infrastructure réseau pour rendre service à un utilisateur.</p> <p>L'objectif de cette séance est :</p> <ul style="list-style-type: none">• d'appréhender la notion de service, de port d'écoute, de connexion client/serveur ;• de connaître les principaux services ainsi que les protocoles associés ;• de paramétrer l'accès à un service et savoir exploiter un service de base. <p>Cette séance se focalise sur les services et les protocoles applicatifs. Les modèles de référence TCP/IP et OSI seront abordés dans la séance suivante.</p>
Savoirs	<p>Savoir-faire</p> <ul style="list-style-type: none">• Exploiter un service de base• Analyser des unités de données de protocole <p>Savoirs associés</p> <ul style="list-style-type: none">• Modèles de référence associés aux architectures réseaux• Services de base et unités de données de protocole associées
Compétences	
Transversalité	
Prérequis	Adressage IP et principes généraux de la commutation.
Outils	PC avec lecture de vidéos possibles. <i>Les vidéos peuvent aussi être consultables à la maison dans le cadre d'une pédagogie de classe inversée.</i>
Mots-clés	IP applications client serveur service port http https ftp smtp pop imap dhcp dns
Durée	6/8 heures
Auteur.e(s)	Apollonie Raffalli et David Duron
Version	v 1.0
Date de publication	Décembre 2017

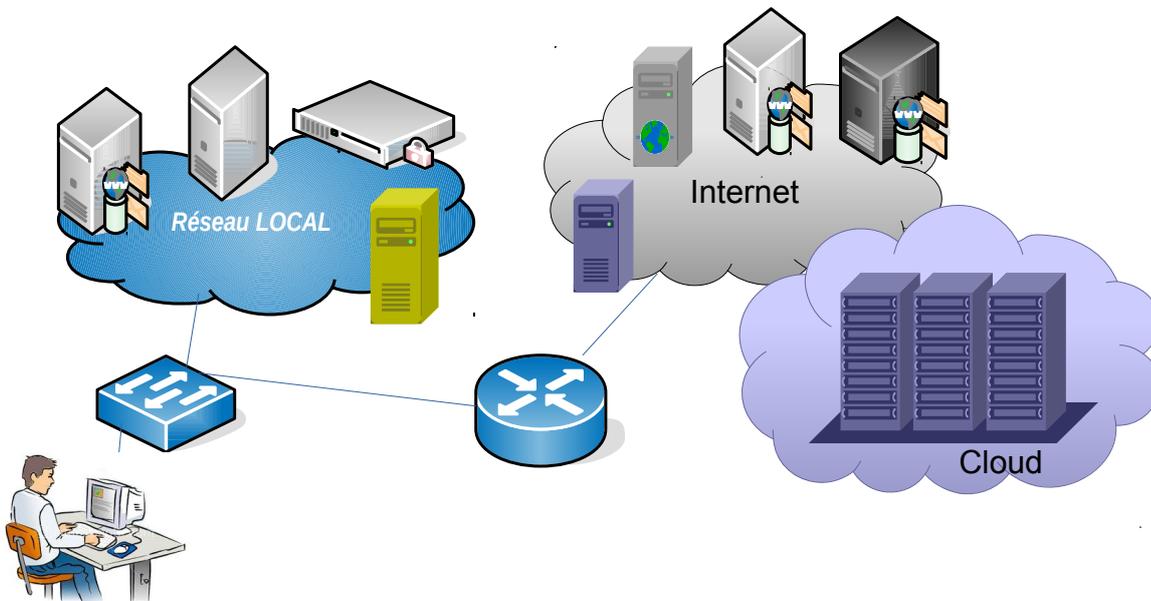
Indications à destination des enseignants

Cette ressource comprend un cours réalisable en classe entière et un QCM de validation. Il est également envisageable :

- de poursuivre la séance avec un TP faisant appel à la commande *netstat* (et nouvellement *ss* sous Linux) et à un scanneur de port (comme *nmap* sur Linux) ;
- d'approfondir les principes de client/serveur avec [l'exolab](#) « Expérimentation du modèle client-serveur par la simulation ». À noter que ces notions qui constituent une petite introduction aux thèmes abordés dans le module commun SI5 seront également approfondies dans ce dernier.

I Les services en ligne et serveurs associés

Des millions de serveurs sont connectés à Internet pour fournir des services. De même, dans la ferme des serveurs de chaque organisation cohabitent plusieurs applications accessibles en ligne à partir d'un poste de travail.



NB : Le cloud computing (ou l'informatique en nuage) est un ensemble de services en ligne avec, généralement, un accès via une interface Web. Ces services sont de plusieurs types, ils offrent notamment :

- du stockage de fichier (par exemple Dropbox ou Google Drive) ;
- de la puissance de calcul de serveurs pour des applications exigeantes (Amazon) ;
- diverses applications permettant de dématérialiser des processus de gestion de l'organisation.

Il existe deux types de cloud, un **public** (Dropbox, Google Drive, etc.) et un **privé** : les serveurs et services sont hébergés dans des centres de données (datacenters) mais sont gérés par l'organisation.

➤ **Visionnez la vidéo** <https://www.youtube.com/watch?v=CiHalbnBgA4>

Q1. Quel est le rôle d'un serveur informatique (vous fournirez un exemple bien connu) ?

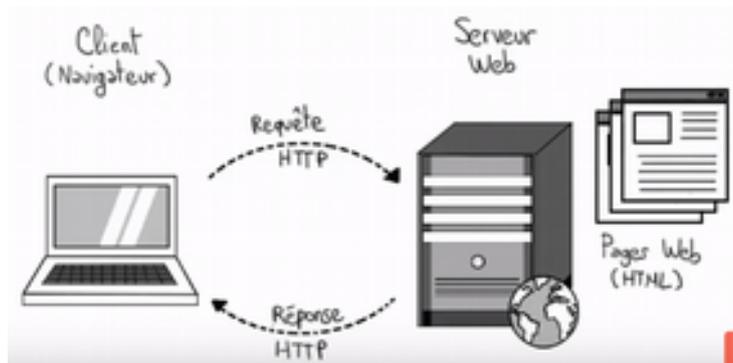
Q2. Citez des services connus offerts sur Internet et dans les organisations.

Q3. Quelles sont les principales différences entre un ordinateur client et un ordinateur serveur ?

II La notion de client/serveur

➤ Visionnez la vidéo <https://www.youtube.com/watch?v=msB9AvJ4bTM>

Q1. Qu'appelle-t-on « logiciel serveur » ?



Q2. Qu'appelle-t-on « logiciel client » ?

1 Services et protocoles de communication

➤ Visionnez la vidéo sur les protocoles de communication: https://www.youtube.com/watch?v=Ds7TvEvM9z4&index=5&list=PLtY_3d8bxqAodgaDtIHr7DY3My7tZx9o1

Q1. Qu'est-ce qu'un protocole de communication et quelle est son utilité ?

Q2. Qu'est-ce qu'un protocole de bas niveau ? Donnez quelques exemples.

Q3. Qu'est-ce qu'un protocole de haut niveau ou protocole applicatif ? Donnez 4 exemples.



Un protocole de haut niveau est donc un protocole portant sur les règles d'organisation de l'information. Dans l'exercice qui suit, nous allons étudier comment est organisée la requête cliente pour qu'elle soit recevable et compréhensible par le serveur et quelles sont les informations contenues dans la réponse, etc.

Exercice – Étude rapide du protocole FTP (file transfert protocol)

Le protocole FTP permet de déposer et de transférer des fichiers d'un serveur. Il est possible d'utiliser un client FTP graphique tel que *Filezilla*. Vous trouverez ci-dessous un exemple de connexion au serveur FTP ftp.funet.fi (que vous pouvez expérimenter) avec le détail de certaines commandes envoyées par le logiciel client selon le protocole FTP. Vous pouvez consulter ici https://fr.wikipedia.org/wiki/Liste_des_codes_des_r%C3%A9ponses_d%27un_serveur_FTP les codes des réponses des serveurs FTP.

Pour l'expérimentation :

- Lancez le client FTP.
- Si vous êtes sur le client Ubuntu, vérifiez (par un clic droit dans la partie haute) que l'option « Afficher les détails du journal » est cochée. Au besoin relancez la connexion.
- Tapez le nom de l'hôte distant, c'est-à-dire celui du serveur FTP.
- Un clic sur « Connexion rapide » choisit automatiquement « anonymous » comme identifiant par défaut, sans mot de passe, et le port 21 comme port par défaut.

The screenshot shows the FileZilla FTP client interface. Several callout boxes provide instructions and labels:

- 1. Taper ici le nom du serveur FTP**: Points to the 'Hôte' field containing 'ftp.funet.fi'.
- 2. Cliquer sur Connexion Rapide**: Points to the 'Connexion rapide' button.
- Partie haute qui affiche le détail de la connexion : Les commandes et leur résultat**: Points to the command log area showing commands like 'TYPE I', 'PASV', 'MLSD' and their corresponding responses.
- Répertoire courant local (actuellement sélectionné)**: Points to the local site tree showing folders like 'david', 'lib', 'lib64', etc.
- Contenu du répertoire local**: Points to the local file list table.
- Contenu du répertoire distant**: Points to the remote site tree showing folders like 'ftp', 'index', 'pub', etc.
- Partie dans laquelle s'affichent les transferts (en cours, achevés, en échec)**: Points to the transfer status area at the bottom.

- Examinez en détail les différentes opérations effectuées et listées dans la partie haute du logiciel, en remontant grâce à « l'ascenseur » vertical.
- Téléchargez un fichier depuis le serveur FTP, par exemple le fichier « README » qui se trouve à la racine de l'arborescence.

Un exemple des commandes affichées est proposé sur la page suivante. Votre affichage peut différer quelque peu en fonction de la version de votre outil client.

Hôte : ftp.funet.fi Identifiant : anonymo Mot de passe : Port : Connexion rapide ▾

Statut : Résolution de l'adresse de ftp.funet.fi
Statut : Connexion à 193.166.3.2:21...
Statut : Connexion établie, attente du message d'accueil...
Réponse : 220----- Welcome to Pure-FTPd [privsep] -----
Réponse : 220-You are user number 46 of 1000 allowed.
Réponse : 220-Local time is now 15:33. Server port: 21.
...
Commande : USER anonymous
Réponse : 331-Welcome to the FUNET anonymous ftp archive
...
Réponse : 331-This is a Dell R710 server with 72GB of RAM. It has two QuadCore
...
Réponse : 331 Any password will work
Commande : PASS *****
Réponse : 230 Any password will work
Commande : SYST
Réponse : 215 UNIX Type: L8
...
Commande : PWD
Réponse : 257 "/" is your current location
...
Lors du téléchargement :
Commande : RETR README
Réponse : 150-Accepted data connection
Réponse : 150 16.6 kbytes to download
Réponse : 226-File successfully transferred
Réponse : 226 0.013 seconds (measured here), 1.27 Mbytes per second
Statut : Transfert de fichier réussi, 17,4 Ko transférés en 1 seconde

Q1. Dites en justifiant si la connexion a été correctement établie.

Q2. Dites en justifiant si la connexion a nécessité l'envoi d'un nom d'utilisateur et d'un mot de passe.

Q3. Expliquez le rôle des commandes SYST, PWD et RETR

2 Les logiciels clients et serveurs

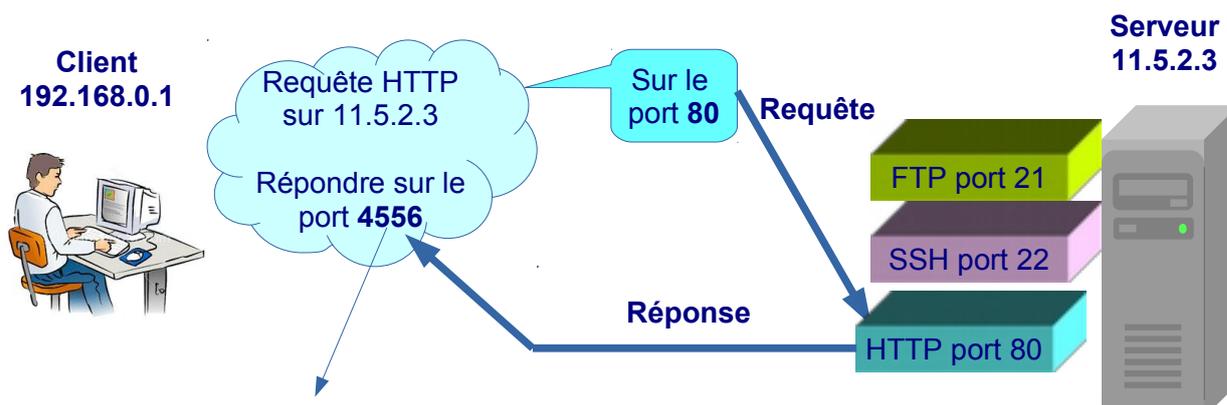
Q1. Complétez le tableau suivant qui donne un ou des exemples de logiciels "client" et de logiciels "serveur" couramment utilisés dans le cadre des services définis.

Service	Serveur	Applications clientes	Applications serveur
Transfert d'une page en HTML correspondante à une URL	Serveur Web		
Envoi et réception de mél	Serveur de messagerie		
Transfert d'un fichier	Serveur FTP		
Administration distante sécurisée d'un serveur	Serveur SSH		
Administration distante non sécurisée d'un serveur	Serveur Telnet		
Fourniture d'une adresse IP correspondant à un nom de domaine	Serveur DNS	Resolver client	Bind
Attribution d'une configuration réseau (adresse IP, masque, DNS, etc)	Serveur DHCP	Processus Client DHCP	DHCPD

3 La cohabitation de plusieurs services sur un même serveur et l'utilité des numéros de port

Sur un serveur disposant d'une adresse IP, plusieurs services (HTTP, FTP, etc.) peuvent cohabiter, mais :

- Comment le serveur « sait-il » à quel service est destinée une requête ?
- Comment un client peut-il recevoir d'un serveur plusieurs réponses à ses requêtes sans que tout ne se mélange ?



1. Le serveur écoute sur un numéro de **port bien défini**.
2. Le client envoie sa requête (selon le protocole) sur ce **port** et génère pour lui-même un numéro de **port aléatoire** sur lequel le serveur pourra lui répondre.
3. Le serveur renvoie la réponse sur le **port aléatoire défini par le client**.



Les ports sont des numéros d'identification qui permettent de spécifier le service concerné. Ce numéro de port est écrit sur 2 octets, ce qui donne 65535 ports possibles (parce que le port 0 n'est, a priori, pas utilisé).

La combinaison "**adresse IP:numéro de port**" constitue ce que l'on appelle une "**socket**" (qui veut dire "connecteur" en anglais) qui identifie pleinement le service qui est concerné sur une machine donnée.

Q2. Complétez le tableau ci-dessous illustrant les ports par défaut des services « connus ».

Service	Serveur	Protocole	Num de port
Transfert d'une page en HTML correspondante à une URL	Serveur Web	HTTP	
Envoi de mél	Serveur de messagerie	SMTP	
Réception de mél	Serveur de messagerie	POP/IMAP	
Transfert d'un fichier	Serveur FTP	FTP	
Administration distante sécurisée d'un serveur	Serveur SSH	SSH	
Administration distante non sécurisée d'un serveur	Serveur Telnet	TELNET	
Fourniture d'une adresse IP correspondant à un nom de domaine	Serveur DNS	DNS	
Attribution d'une configuration réseau (adresse IP, masque, DNS, etc)	Serveur DHCP	DHCP	

Les numéros de port **entre 0 et 1023 sont réservés aux applications standards**. On appelle généralement cela les **Well-Known** ports. Ces numéros sont attribués par l'IANN.

Les numéros de port **entre 1024 et 49151** sont des **ports inscrits** (ou ports enregistrés) c'est-à-dire qu'ils sont réservés pour des applications couramment utilisées (comme le port 3306 pour le serveur MySQL).

Sur Linux, vous pouvez trouver dans le fichier `/etc/services` une liste des ports *Well-Known* et de certains ports inscrits (avec le nom des services correspondants).

Sur Windows, l'emplacement de ce fichier est (par défaut) : `C:\Windows\System32\drivers\etc\services`.

Les numéros de port de **49151 à 65535** sont des **ports dynamiques** (et/ou privés) utilisés généralement en port source par les applications clientes pour se connecter à un serveur.



Les protocoles applicatifs ont généralement leur équivalent sécurisé (notamment transfert des données chiffrées et authentification du destinataire et/ou de l'émetteur) : port 443 pour le protocole HTTPS, ports 587, 465, 587 pour SMTP, 995 pour POP, etc.

La commande *netstat* (sur Linux* et sur Windows) permet de visualiser l'état des connexions :

```
root@servDebian:~# netstat -taun
Proto Recv-Q Send-Q Local Address Foreign Address State
tcp 0 0 0.0.0.0:22 0.0.0.0:* LISTEN
tcp 0 0 127.0.0.1:25 0.0.0.0:* LISTEN
tcp 0 0 127.0.0.1:3306 0.0.0.0:* LISTEN
tcp 0 0 10.22.60.100:22 172.16.160.100:52987 ESTABLISHED
```

* Cette commande est en fait obsolète sur Linux et a été remplacée par la commande « *ss* » (qui fonctionne de la même manière) sur les distributions récentes.

Q3. Quels sont les services en écoute sur la machine "servDebianSIO1" (state LISTEN) ?

Q4. Quelle est la connexion établie sur la machine "servDebian" (state ESTABLISHED) ? Vous préciserez l'adresse IP et le port du client ainsi que l'adresse IP et le port du serveur.