

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

Couche application

Les protocoles d'application

Luc Lavoie
Département d'informatique
Faculté des sciences
luc.lavoie@USherbrooke.ca

IFT585 Télématique (T10-Application_v093a) 1

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

Remerciements

- Les présentes diapositives ont été initialement développées par les auteurs de
 - **Computer Networking:
A Top Down Approach Featuring the Internet,
2nd edition.
Jim Kurose, Keith Ross
Addison-Wesley, July 2002.**
- Elles ont été traduites, adaptées, modifiées et complétées par Luc Lavoie qui assume la responsabilité entière des éventuelles erreurs.
- Les auteurs originaux demandent (et c'est la moindre des choses), qu'advenant une utilisation du matériel en classe, les sources soient citées.
- Ils demandent aussi que la diffusion sous quelque forme que ce soit (site web, photocopie ou autres) soit accompagnée de la mention du copyright.
- Copyright 1996-2002, J.F Kurose and K.W. Ross, All Rights Reserved
- Finalement, j'aimerais remercier ici les auteurs de leur généreuse contribution.

LL, 2007-10-24 et 2008-01-15

IFT585 Télématique (T10-Application_v093a) 2

UNIVERSITÉ DE
SHERBROOKE

2009-03-28


A note on the use of these ppt slides

We're making these slides freely available to all (faculty, students, readers). They're in powerpoint form so you can add, modify, and delete slides (including this one) and slide content to suit your needs. They obviously represent a *lot* of work on our part. In return for use, we only ask the following:

- ❑ If you use these slides (e.g., in a class) in substantially unaltered form, that you mention their source (after all, we'd like people to use our book!)
- ❑ If you post any slides in substantially unaltered form on a www site, that you note that they are adapted from (or perhaps identical to) our slides, and note our copyright of this material.

Thanks and enjoy! JFK/KWR

All material copyright 1996-2002
J.F Kurose and K.W. Ross, All Rights Reserved



Computer Networking: A Top Down Approach Featuring the Internet, 2nd edition.
Jim Kurose, Keith Ross
Addison-Wesley, July 2002.

IFT585 Télématique (T10-Application_v093a) 3

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

Objectifs

- ❑ Introduction à la conception de protocoles d'application
 - utilisation des services de transport
 - modèle client-serveur
 - modèle poste-à-poste
- ❑ Survol de quelques protocoles courants
 - HTTP, URL
 - DNS
 - Telnet
 - FTP
 - SMTP, POP3, IMAP, MIME
 - SSH, SSL, S/MIME
 - RTP, SIP
 - ...

IFT585 Télématique (T10-Application_v093a) 4

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Sommaire

- **Principes généraux**
 - clients et serveurs
 - exigences
- **Protocoles courants**
 - HTTP, URL
 - DNS
 - Telnet
 - FTP
 - SMTP, POP3, IMAP, MIME
 - SSH, SSL, S/MIME
 - RTP, SIP
 - ...
- **Serveur Web**
 - Conception
 - Programmation
- **Diffusion de contenu**
 - « Web caching »
 - Réseaux de diffusion de contenu
 - Partage de fichiers P2P

IFT585 Télématique (T10-Application_v093a) 5

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Application en réseaux un peu de vocabulaire - **Processus**

Programme exécuté sur un équipement

- sur un même équipement, les processus peuvent communiquer entre eux par divers mécanismes définis par le SE
 - par exemple : mémoire partagée, sémaphores, rendez-vous, fils d'exécution, ...
- sur des équipements reliés par un réseau, les processus peuvent communiquer entre eux par un protocole d'application
 - note : un équipement peut être en réseau avec lui-même !

IFT585 Télématique (T10-Application_v093a) 6

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

Tendances (LL)

- ❑ Plusieurs chercheurs militent en faveur de l'unification des modes de communication entre processus (au niveau applicatif).
- ❑ CSP (Hoare) en est une illustration très convaincante et déjà ancienne.
- ❑ Le principal frein est l'immobilisme des systèmes d'exploitation : il n'y pas eu d'innovation (commerciale) depuis plus de 20 ans.

IFT585 Télématique (T10-Application_v093a) 7

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

Application en réseaux un peu de vocabulaire - **Agent**

Interface entre l'utilisateur et le réseau

- IPM (interface personne-machine)
- protocole d'application

Exemples

- Fureteur web (Firefox, Explorer, Safari, Chrome)
- Agent de courriel (Outlook, Mail, Eudora)
- Lecteur multimédia (Media Player, Real Player, iTunes)

IFT585 Télématique (T10-Application_v093a) 8

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Protocoles et protocoles d'application

Application
→ processus répartis

- sur plusieurs équipements
- échangeant des messages
- pour offrir leurs services

Protocole d'application

- est un composant de l'application
- définit les messages échangés par les applications et les actions qui en découlent
- utilise les services de la couche de transport
- intègre vraisemblablement des services internes de présentation et de session

IFT585 Télématique (T10-Application_v093a) 9

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Un protocole (d'application) comprend

- Types de messages échangés (requêtes, réponses, ...)
- Syntaxe des messages (structure et composition)
- Sémantique des messages
- Règles d'échange des messages

Protocoles publics

- assurent l'interopérabilité
- définis par des organismes indépendants et reconnus
 - ISO
 - IUT (CCITT)
 - IETF (RFC -> STD)
 - IEEE
- exemples
 - HTTP, SMTP, ...

Protocoles privés

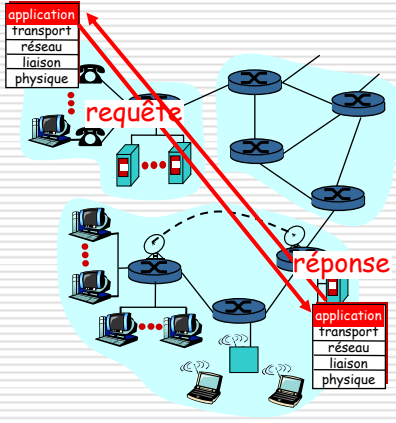
- exemples
 - KaZaA, Skype, ...

IFT585 Télématique (T10-Application_v093a) 10

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Modèle client-serveur

- Client**
 - a besoin d'un service
 - prend contact avec le serveur (« parle d'abord »)
 - envoie une requête
- Serveur**
 - fournit un service
 - attend les requêtes (de service) des clients
 - retourne une réponse
- Exemples**
 - service web
 - service de courriel



IFT585 Télématique (T10-Application_v093a) 11

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Modèle poste-à-poste

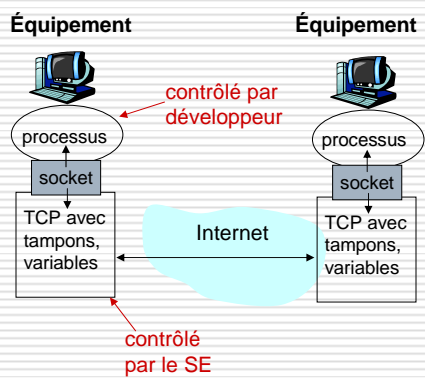
- Qu'est-ce ?**
 - Le mode normal, sans spécialisation des équipements !
- Pourquoi en parle-t-on autant ?**
 - Parce qu'il y a désormais plusieurs embûches
 - architectures de réseaux
 - NAT (pénurie d'adresse)
 - PROXY (pénurie de bande passante)
 - volonté de contrôle et de centralisation
 - sécurité
 - cohésion

IFT585 Télématique (T10-Application_v093a) 12

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Communication grâce à un réseau

- Un processus envoie et reçoit les messages par l'entremise de « sockets » (points de service).
- Un socket est une « porte »
 - processus émetteur « sort » le message
 - processus émetteur assure l'infrastructure de transport de chaque côté de la porte qui transporte le message au socket du processus récepteur
- API (« application programming interface »)
 - choix du protocole de transport
 - possibilité de contrôler certains paramètres
 - (sera élaboré plus loin)



IFT585 Télématique (T10-Application_v093a) 13

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Adressage de la couche de transport (rappel)

- Pour qu'un processus puisse recevoir des messages, il doit avoir un identificateur
- Chaque équipement a une adresse IP unique de 32-bit
- **Question :**
L'adresse IP de l'équipement qui exécute le processus est-elle suffisante pour identifier le processus ?
Réponse :
Non, un même équipement peut exécuter plusieurs processus.
- L'identificateur est composé de l'adresse IP et des **numéros de port** associés au processus sur l'équipement.
- Exemple de numéros de port :
 - Serveur HTTP : 80
 - Serveur de courrier : 25
- Sera élaboré plus loin

IFT585 Télématique (T10-Application_v093a) 14

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Adressage applicatif (constat)

- Celui de la couche transport est totalement insuffisant
 - multiplicité des applications pour une même adresse
 - couplage abusif application-équipement

IFT585 Télématique (T10-Application_v093a) 15

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Exigences applicables aux couches inférieures (transport et réseau)

Fiabilité (perte de données)

- Certaines applications peuvent tolérer des pertes
 - audio, vidéo
- Certaines autres applications nécessitent un transfert de données à 100% fiable
 - transfert de fichiers, telnet

Délai (Timing)

- Certaines applications nécessitent des délais courts pour être « efficaces »
 - téléphonie Internet
 - jeux interactifs

Bande passante

- Certaines applications ont besoin d'une bande passante minimale pour être « efficaces »
 - multimédia
- Certaines autres applications utilisent la bande passante qui est disponible (applications « élastiques »)
 - mises à jour

et plusieurs autres exigences qui, ensemble, forment la « qualité de service »

IFT585 Télématique (T10-Application_v093a) 16

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Exemples d'exigences de certaines applications courantes

Application	Perte de données	Bande passante	Décalage temporel
transfert de fichiers	aucune perte acc.	élastique	non
courriel	aucune perte acc.	élastique	non
documents web	aucune perte acc.	élastique	non
audio/vidéo en temps réel	résistant aux pertes	audio: 5kbps-1Mbps vidéo: 10kbps-5Mbps	oui (n x 100 msec)
audio/vidéo stockés	résistant aux pertes	comme ci-dessus	oui (n sec)
jeux interactifs	résistant aux pertes	variable	oui (n x 100 msec)
Messagerie instantanée	aucune perte acc.	faible	oui (et non!) avec $1 \leq n \leq 9$

IFT585 Télématique (T10-Application_v093a) 17

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Services offerts par les protocoles de transport

<p>Services TCP</p> <ul style="list-style-type: none"> <input type="checkbox"/> Transfert : évidemment! <input type="checkbox"/> Connexion : consensus d'ouverture, de maintien et de fermeture de la communication <input type="checkbox"/> Fiabilité : intégrité des données transmises <input type="checkbox"/> Contrôle de flot : l'émetteur n'écrasera pas le récepteur <input type="checkbox"/> Contrôle de congestion : l'émetteur ne surchargera pas le réseau <input type="checkbox"/> Ne fournit pas : la garantie de délais, la garantie de bande passante, la qualité de service en général 	<p>Services UDP</p> <ul style="list-style-type: none"> <input type="checkbox"/> Transfert : transfert de données non fiable entre le processus émetteur et le processus récepteur <input type="checkbox"/> Rien d'autre ! <ul style="list-style-type: none"> ■ pas de connexion ■ pas de fiabilité ■ pas de contrôle de flot ■ pas de contrôle de congestion <input type="checkbox"/> Question Pourquoi y a-t-il un UDP ?
--	---

IFT585 Télématique (T10-Application_v093a) 18

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Applications Internet : protocoles d'application et de transport

Application	Protocole applicatif	Protocole de transport
courriel	SMTP [RFC 2821]	<allez voir!>
commande à distance	Telnet [RFC 854]	UDP
web	HTTP [RFC 2616]	TCP
transfert de fichiers	FTP [RFC 959]	TCP
streaming multimédia	privé (RealNetworks, ...)	TCP TCP
téléphonie internet	privé (Skype, ...)	TCP ou UDP UDP (surtout)

IFT585 Télématique (T10-Application_v093a) 19

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Vocabulaire

- ❑ **Flux de données** (data stream)
 - Séquence continue de données qui sont transmises d'un point à un autre dans un réseau de télécommunication.
 - Couches ???
- ❑ **Flot de données** (data flow)
 - Regroupement d'informations qui circulent dans un réseau de télécommunication, depuis leur point d'origine jusqu'à leur destination.
 - Couches ???

Merci GDT !

IFT585 Télématique (T10-Application_v093a) 20

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Sommaire

- Principes généraux
 - clients et serveurs
 - exigences
- Protocoles courants
 - **HTTP, URL**
 - DNS
 - Telnet
 - FTP
 - SMTP, POP3, IMAP, MIME
 - SSH, SSL, S/MIME
 - RTP, SIP
 - ...
- Serveur Web
 - Conception
 - Programmation
- Diffusion de contenu
 - « Web caching »
 - Réseaux de diffusion de contenu
 - Partage de fichiers P2P

IFT585 Télématique (T10-Application_v093a) 21

UNIVERSITÉ DE SHERBROOKE 2009-03-28

HTTP : HyperText Transport Protocol

- Origine
 - CERN
- Objectifs
 - affichage d'informations textuelles et graphiques mises à jour fréquemment
 - mécanisme de publication accessible sans programmation ni application spécialisée
 - application d'accès (navigateur) minimale et très portable
- Inspiration
 - Minitel

IFT585 Télématique (T10-Application_v093a) 22

UNIVERSITÉ DE SHERBROOKE 2009-03-28

HTTP : Vocabulaire

- Une page Web est composée d'objets
- Un objet peut être un fichier HTML, une image JPEG, un applet Java, un fichier audio, etc.
- Une page Web est un document HTML de base qui fait référence à des objets
- Chaque objet peut être référencé à l'aide d'une URL
- Exemple d'URL :

`www.uneEcoleX.org/unDepartement/logo.gif`

 Nom de domaine chemin

IFT585 Télématique (T10-Application_v093a) 23

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Adressage (de la confusion service et équipement)

Un nom de domaine est-il un nom d'équipement ?

- en fait, le nom de domaine est un emplacement de service... et non forcément un équipement!

IFT585 Télématique (T10-Application_v093a) 24

UNIVERSITÉ DE SHERBROOKE 2009-03-28

HTTP : Survol

HTTP :
(hypertext transfer protocol)

- Protocole de transfert hypertexte
- Modèle client-serveur
 - **client** : l'agent émet les requêtes puis reçoit et affiche les objets d'une page hypertexte
 - **serveur** : serveur répond aux requêtes et envoie les objets de la page demandée
- HTTP 1.0: RFC 1945
- HTTP 1.1: RFC 2068

IFT585 Télématique (T10-Application_v093a) 25

UNIVERSITÉ DE SHERBROOKE 2009-03-28

HTTP : Survol (suite)

TCP :

- Le client ouvre la connexion TCP (crée le socket vers le port 80 du serveur)
- Le serveur accepte la demande de connexion TCP du client
- Les messages HTTP sont échangés entre le navigateur (client HTTP) et le serveur (serveur HTTP)
- Le client et le serveur ferment la connexion TCP

HTTP est « sans état »

- Le serveur ne conserve pas l'historique des requêtes du client

de plus

Les protocoles qui conservent les états sont complexes

- L'historique (état) doit être maintenu
- Si le serveur/client plante, les vues des états peuvent être incohérentes et doivent être reconstruites.

par contre... (à suivre)

IFT585 Télématique (T10-Application_v093a) 26

UNIVERSITÉ DE SHERBROOKE 2009-03-28

HTTP : Quelques constats

- Bonne adéquation par rapport aux objectifs d'origine
- L'utilisation de messages humainement interprétables a grandement contribué au développement rapide d'applications et à la diffusion du protocole
- Performance médiocre (longs délais)
- Absence d'état handicape grandement le développement d'applications « réelles »

IFT585 Télématique (T10-Application_v093a) 27

UNIVERSITÉ DE SHERBROOKE 2009-03-28

HTTP : Connexions

<p><u>HTTP non persistant</u></p> <ul style="list-style-type: none"> <input type="checkbox"/> Au moins un objet est envoyé via la connexion TCP. <input type="checkbox"/> HTTP/1.0 utilise HTTP non persistant 	<p><u>HTTP persistant</u></p> <ul style="list-style-type: none"> <input type="checkbox"/> Plusieurs objets peuvent être envoyés grâce à une même connexion TCP entre le client et le serveur <input type="checkbox"/> HTTP/1.1 utilise les connexions persistantes par défaut
--	---

IFT585 Télématique (T10-Application_v093a) 28

UNIVERSITÉ DE SHERBROOKE 2009-03-28

HTTP non persistant

Supposons que l'utilisateur inscrit l'URL `www.uneEcoleX.org/unDepartement/home.index` (contient du texte et fait référence à 10 images jpeg)

- 1a. Le client HTTP ouvre la connexion TCP sur le port 80 du serveur HTTP (processus) du site `www.uneEcoleX.org`
- 1b. Le serveur HTTP de l'équipement `www.uneEcoleX.org` attend la connexion TCP sur le port 80, « accepte » la connexion, avise le client
2. Le client HTTP envoie le *message requête* HTTP (qui contient l'URL) au socket de la connexion TCP. Le message indique que le client veut un objet `unDepartement/home.index`
3. Le serveur HTTP reçoit le message requête, construit le *message réponse* qui contient l'objet requis et envoie le message dans le socket

temps

IFT585 Télématique (T10-Application_v093a) 29

UNIVERSITÉ DE SHERBROOKE 2009-03-28

HTTP non persistant (suite)

- 4a. Le serveur HTTP demande de fermer la connexion TCP.
- 5a. Le client HTTP reçoit le message réponse qui contient le fichier html, affiche la page html. Il analyse le fichier html et y trouve les références à 10 objets jpeg
- 5b. Le client HTTP demande de fermer la connexion TCP
- 4b. Le serveur HTTP ferme la connexion TCP.
6. Répète les étapes 1 à 5 pour chacun des 10 objets jpeg

temps

IFT585 Télématique (T10-Application_v093a) 30

UNIVERSITÉ DE SHERBROOKE 2009-03-28

HTTP non persistant : Calcul du temps de réponse

RTT : temps de transmission d'un petit paquet entre le client et le serveur, aller-retour.

Temps de réponse :

- un RTT pour ouvrir la connexion TCP
- un RTT pour une requête HTTP et les premiers octets de la réponse HTTP retournée
- TT: temps de transmission d'un fichier

total = 2*RTT+TT

IFT585 Télématique (T10-Application_v093a) 31

UNIVERSITÉ DE SHERBROOKE 2009-03-28

HTTP persistant

Problèmes du HTTP non persistant

- requiert 2 RTT par objet
- SE doit allouer les ressources de l'équipement pour chaque connexion TCP
- mais les navigateurs ouvrent souvent des connexions TCP en parallèle pour aller chercher les objets référencés.

HTTP persistant

- le serveur laisse la connexion ouverte après l'envoi de la réponse
- les messages HTTP entre les mêmes client/serveur subséquents sont envoyés via cette connexion

Persistant sans pipelining

- le client émet une nouvelle requête seulement lorsque la réponse de la requête précédente a été reçue
- un RTT pour chaque objet référencé

Persistant avec pipelining

- par défaut dans HTTP/1.1
- le client émet une nouvelle requête dès qu'il rencontre des objets référencés
- ultimement, un RTT pour l'ensemble des objets référencés

IFT585 Télématique (T10-Application_v093a) 32

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Message HTTP de type requête

- ❑ deux types de messages HTTP : *requête*, *réponse*
- ❑ requête HTTP :
 - ASCII (format interprétable par l'utilisateur)

ligne de la requête (commandes GET, POST, HEAD) → GET /dossier/page.html HTTP/1.1

lignes d'entête → Host: www.uneEcoleX.org
User-agent: Mozilla/4.0
Connection: close
Accept-language: fr

ligne vide indique la fin du message → (fin de ligne supplémentaire : CR+LF)

IFT585 Télématique (T10-Application_v093a) 33

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Format d'un message HTTP de type requête

method	sp	URL	sp	version	cr	lf	request line
header field name	:	value	cr	lf			
		•					
		•					
		•					
header field name	:	value	cr	lf			} header lines
cr	lf						
Entity Body							

IFT585 Télématique (T10-Application_v093a) 34

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Deux techniques de téléchargement d'un formulaire de saisie

Technique Post

- Les pages web contiennent souvent des formulaires de saisie
- Les données du formulaire sont téléchargées vers le serveur dans la charge utile du message (EntityBody)

Technique URL

- Utilise la méthode GET
- Les données du formulaire sont téléchargées dans le champ URL de la ligne de requête
- Exemple
 - `www.zoo.com/animal?castor&bouleau`

IFT585 Télématique (T10-Application_v093a) 35

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Types de méthodes

<h3><u>HTTP/1.0</u></h3> <ul style="list-style-type: none"> <input type="checkbox"/> GET <input type="checkbox"/> POST <input type="checkbox"/> HEAD <ul style="list-style-type: none"> ■ même réponse que GET, mais sans la charge utile (EntityBody) 	<h3><u>HTTP/1.1</u></h3> <ul style="list-style-type: none"> <input type="checkbox"/> GET, POST, HEAD <input type="checkbox"/> PUT <ul style="list-style-type: none"> ■ télécharge le contenu de la charge utile vers le « conteneur » spécifié dans le champ URL <input type="checkbox"/> DELETE <ul style="list-style-type: none"> ■ supprime le « conteneur » spécifié dans le champ URL
---	---

IFT585 Télématique (T10-Application_v093a) 36

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Message HTTP de type réponse

ligne de statut
(protocole
code de statut
texte du statut)

lignes d'entête

```
HTTP/1.1 200 OK
Connection: close
Date: Thu, 06 Aug 1998 12:00:15 GMT
Server: Apache/1.3.0 (Unix)
Last-Modified: Mon, 22 Jun 1998 .....
Content-Length: 6821
Content-Type: text/html
```

données (par exemple,
un texte HTML,
un fichier CSS, etc.)

```
données données données données ...
```

IFT585 Télématique (T10-Application_v093a) 37

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Codes de statut des messages : HTTP de type réponse

Dans la 1^{re} ligne du message réponse serveur->client.
Exemples de codes :

- ❑ **200 OK**
 - Requête traitée avec succès, objet requis joint au message
- ❑ **301 Moved Permanently**
 - Objet requis déplacé, nouvel emplacement indiqué sous le paramètre « Location: »
- ❑ **400 Bad Request**
 - Requête considérée non conforme par le serveur
- ❑ **404 Not Found**
 - Le document requis n'a pas été trouvé sur le serveur
- ❑ **505 HTTP Version Not Supported**

IFT585 Télématique (T10-Application_v093a) 38

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Exercice HTTP (côté client)

1. Telnet à votre serveur Web favori ☺
`telnet pages.USherbrooke.ca 80`

Ouvre une connexion TCP sur le port 80 (port par défaut du serveur HTTP) sur pages.USherbrooke.ca

2. Soumettre une requête HTTP GET
`GET /llavoie/index HTTP/1.0`

Lorsque cette commande est envoyée (appuyer sur la touche <Entrée> 2 fois), cette requête GET (minimale mais complète) est envoyée au serveur HTTP

3. Consulter le message envoyé par le serveur HTTP !

IFT585 Télématique (T10-Application_v093a) 39

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Le problème de l'autorisation... en l'absence d'état

Autorisation : gère l'accès au contenu du serveur

- ❑ **Informations d'autorisation :** typiquement nom, mot de passe
- ❑ **Sans statut :** le client doit fournir les informations d'autorisation dans *chaque* requête
 - **autorisation :** ligne d'entête de chaque requête
 - si aucune **autorisation** en entête: serveur refuse l'accès et envoie :
 WWW authenticate:
 ligne d'entête de la réponse

client	serveur
requête http courante	→
401: authorization req.	←
WWW authenticate:	
requête http courante + Authorization: <cred>	→
réponse http courante	←
requête http courante + Authorization: <cred>	→
réponse http courante	←

temps ↓

IFT585 Télématique (T10-Application_v093a) 40

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Conserver « l'état » : les cookies

Quatre composants

1. cookie émis dans la réponse HTTP
2. cookie retourné dans la requête HTTP
3. fichier cookie stocké sur l'équipement de l'utilisateur et géré par le navigateur de l'utilisateur
4. information associée au cookie dans la BD du serveur

Exemple

- Sylvie se connecte toujours à Internet à partir du même poste de travail.
- Elle consulte un certain site de commerce électronique pour la première fois.
- Lorsque la requête HTTP initiale arrive au site, le site crée un ID unique et ajoute une entrée pour cet ID dans la BD.
- Le serveur transmet l'ID dans un cookie au client.
- Le client conserve le cookie et le joint aux demandes ultérieures.

IFT585 Télématique (T10-Application_v093a) 41

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Conserver « l'état » : les cookies (1)

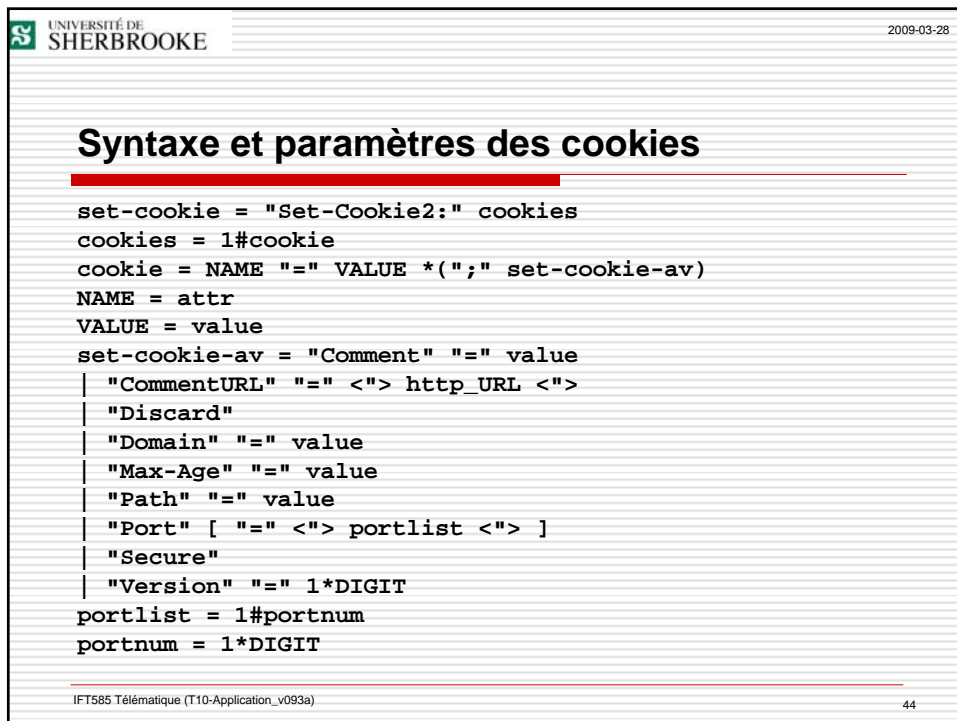
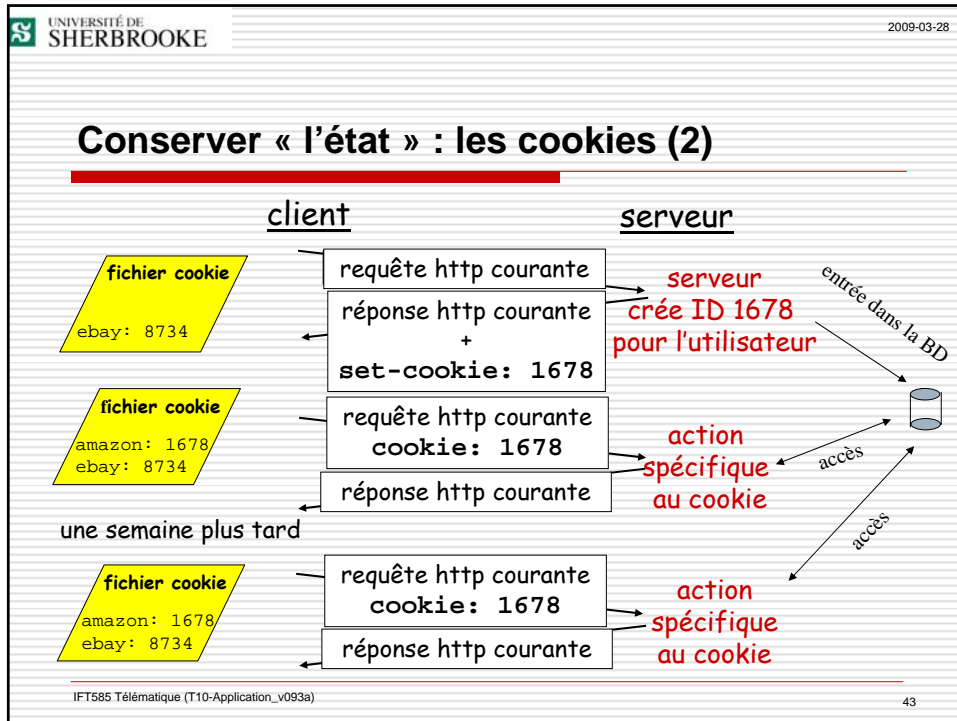
Exemples d'utilisation

- autorisation d'accès
- paniers d'achat virtuel
- paramètres de consultation
- état de la session de l'utilisateur (courriel)

Cookies et confidentialité

- Les cookies permettent aux sites d'obtenir des informations sur leurs visiteurs.
- Les utilisateurs fournissent parfois leur nom et leur adresse courriel aux sites.
- Les moteurs de recherche utilisent la redirection et les cookies pour obtenir plus d'infos.
- Les compagnies publicitaires obtiennent des informations via les sites.

IFT585 Télématique (T10-Application_v093a) 42



UNIVERSITÉ DE SHERBROOKE 2009-03-28

Exemple de cookie (RFC 2965)

- ❑ **1. Agent -> Serveur**
- ❑ POST /acme/login HTTP/1.1
- ❑ [form data]
- ❑ User identifies self via a form.

- ❑ **2. Serveur -> Agent**
- ❑ HTTP/1.1 200 OK
- ❑ Set-Cookie2: Customer="WILE_E_COYOTE"; Version="1"; Path="/acme"
- ❑ Cookie reflects user's identity.

- ❑ **3. Agent -> Serveur**
- ❑ POST /acme/pickitem HTTP/1.1
- ❑ Cookie: \$Version="1"; Customer="WILE_E_COYOTE"; \$Path="/acme"
- ❑ [form data]
- ❑ User selects an item for "shopping basket".

- ❑ **4. Serveur -> Agent**
- ❑ HTTP/1.1 200 OK
- ❑ Set-Cookie2: Part_Numbers="Rocket_Launcher_0001"; Version="1";
- ❑ Path="/acme"
- ❑ Shopping basket contains an item.

- ❑ **5. Agent -> Serveur**
- ❑ POST /acme/shipping HTTP/1.1
- ❑ Cookie: \$Version="1";
- ❑ Customer="WILE_E_COYOTE"; \$Path="/acme";
- ❑ Part_Numbers="Rocket_Launcher_0001"; \$Path="/acme"
- ❑ [form data]
- ❑ User selects shipping method from form.

- ❑ **6. Serveur -> Agent**
- ❑ HTTP/1.1 200 OK
- ❑ Set-Cookie2: Shipping="FedEx"; Version="1"; Path="/acme"
- ❑ New cookie reflects shipping method.

- ❑ **7. Agent -> Serveur**
- ❑ POST /acme/process HTTP/1.1
- ❑ Cookie: \$Version="1";
- ❑ Customer="WILE_E_COYOTE"; \$Path="/acme";
- ❑ Part_Numbers="Rocket_Launcher_0001"; \$Path="/acme";
- ❑ Shipping="FedEx"; \$Path="/acme"
- ❑ [form data]
- ❑ User chooses to process order.

- ❑ **8. Serveur -> Agent**
- ❑ HTTP/1.1 200 OK
- ❑ Transaction is complete.

IFT585 Télématique (T10-Application_v093a) 45

UNIVERSITÉ DE SHERBROOKE 2009-03-28

GET conditionnel : cache côté du client

client
serveur

But : ne pas envoyer l'objet si la version dans le cache du client est à jour

- ❑ **client :** spécifie dans la requête HTTP la date de l'objet dans le cache :
If-modified-since: <date>
- ❑ **serveur :** si la version de l'objet dans le cache est à jour le serveur n'envoie pas l'objet dans la réponse :
HTTP/1.0 304 Not Modified

Requête HTTP
If-modified-since: <date>

→ objet non modifié

Réponse HTTP
HTTP/1.0
304 Not Modified

Requête HTTP
If-modified-since: <date>

→ objet modifié

Réponse HTTP
HTTP/1.0 200 OK
<données>

IFT585 Télématique (T10-Application_v093a) 46

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

Sommaire

- Principes généraux
 - clients et serveurs
 - exigences
- Protocoles courants
 - HTTP, URL
 - **DNS**
 - Telnet
 - FTP
 - SMTP, POP3, IMAP, MIME
 - SSH, SSL, S/MIME
 - RTP, SIP
 - ...
- Serveur Web
 - Conception
 - Programmation
- Diffusion de contenu
 - « Web caching »
 - Réseaux de diffusion de contenu
 - Partage de fichiers P2P

IFT585 Télématique (T10-Application_v093a) 47

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

DNS : Domain Name System

- Humains
 - nom, prénom
 - adresse de résidence
 - numéro de téléphone
 - numéro de passeport
 - matricule étudiant, ...
- Équipements
 - adresse physique (MAC: 1A:78:9B:00:25:48)
 - adresse logique (IP: 132.140.12.5)
 - adresse applicative (gaia.cs.umass.edu)
- Pourquoi des adresses applicatives ?
 - le folklore (les humains ne peuvent mémoriser des chiffres)
 - la motivation réelle (la mobilité par l'indirection)

IFT585 Télématique (T10-Application_v093a) 48

UNIVERSITÉ DE SHERBROOKE 2009-03-28

DNS : L'essence

- *base de données répartie (hiérarchiquement)* sur plusieurs *serveurs de noms*
- *protocole d'application* les équipements, les aiguilleurs et les serveurs de noms interagissent pour traiter les demandes de traduction (« résolution ») de nom
 - note : sans la mise en œuvre effective de ce protocole, l'Internet ne peut être fonctionnel
 - le traitement (et la complexité) est au-dessus des sous-réseaux

IFT585 Télématique (T10-Application_v093a) 49

UNIVERSITÉ DE SHERBROOKE 2009-03-28

DNS : Les serveurs de noms

Pourquoi ne pas centraliser DNS?

- point de défaillance unique
- trafic
- maintenance
- mise à l'échelle

Aucun serveur ne peut détenir toutes les traductions

- serveur de noms local (*local name server*)
 - Chaque nœud (organisation, FSI, etc.) en possède (au moins) un.
 - Toutes les demandes locales sont d'abord desservies localement.
- serveur de noms de référence (*authoritative name server*)
 - conserve les déclarations de référence et est redevable de celles-ci;
 - peut entreprendre des recherches pour le bénéfice d'un tiers.

IFT585 Télématique (T10-Application_v093a) 50

UNIVERSITÉ DE SHERBROOKE 2009-03-28

DNS : Les serveurs « racine »

- ❑ au service (principalement) des serveurs locaux
- ❑ fonctions offertes
 - assurent les requêtes auprès des serveurs de référence
 - obtiennent les correspondances (« mappings »)
 - fournissent les traductions aux serveurs de noms

13+ serveurs « racine »

IFT585 Télématique (T10-Application_v093a) 51

UNIVERSITÉ DE SHERBROOKE 2009-03-28

DNS : Exemple 1

- ❑ surf.eurecom.fr désire la traduction de gaia.cs.umass.edu
- ❑ contacte le serveur local dns.eurecom.fr
- ❑ au besoin, dns.eurecom.fr contacte le serveur racine
- ❑ au besoin, le serveur racine contacte le serveur de référence dns.umass.edu

IFT585 Télématique (T10-Application_v093a) 52

UNIVERSITÉ DE SHERBROOKE 2009-03-28

DNS : Exemple 2

Serveur « racine » :

- peut ne pas connaître le serveur de référence du domaine
- il connaît cependant le serveur de référence d'au moins un domaine hiérarchiquement plus élevé

IFT585 Télématique (T10-Application_v093a) 53

UNIVERSITÉ DE SHERBROOKE 2009-03-28

DNS : Exemple 3

requête récursive

- la charge est placée sur le serveur racine
- que faire s'il est surchargé ?

requête itérée

- le serveur racine répond par un serveur de référence à qui il délègue la recherche

IFT585 Télématique (T10-Application_v093a) 54

UNIVERSITÉ DE SHERBROOKE 2009-03-28

DNS : Mise en cache et mise à jour

- lorsqu'un serveur de noms obtient une information, il la conserve en cache
 - nécessité d'une temporisation (de l'ordre de ...)
- un protocole de mise à jour a aussi été établi par l'IETF
 - voir RFC 2136
 - <http://www.ietf.org/html.charters/dnsind-charter.html>

IFT585 Télématique (T10-Application_v093a) 55

UNIVERSITÉ DE SHERBROOKE 2009-03-28

DNS : Les messages

DNS : base de données distribuée qui stocke les enregistrements des ressources (*resource records*, RR)

format de RR : (name, value, type, ttl)

- **Type=A**
 - **name** : nom de l'équipement
 - **value** : adresse IP
- **Type=NS**
 - **name** : nom du domaine
 - **value** : adresse IP du serveur de référence de ce domaine
- **Type=CNAME**
 - **name** : alias du nom « canonique » (le nom réel)
www.ibm.com est en fait east.backup2.ibm.com
 - **value** : nom « canonique »
- **Type=MX**
 - **value** : nom du serveur SMTP associé à **name**

IFT585 Télématique (T10-Application_v093a) 56

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Protocole DNS, messages

Protocole DNS :
 les messages *requête* et *réponse* ont le même *format de message*

Entête du message

□ **identification :**
 la requête a un numéro de 16 bit, la réponse à la requête utilise le même numéro.

□ **flags :**

- requêtes présentes
- réponses présentes
- récursion souhaitée
- récursion disponible
- réponse de référence

identification	flags	↑ 12 bytes ↓
number of questions	number of answer RRs	
number of authority RRs	number of additional RRs	
questions (variable number of questions)		
answers (variable number of resource records)		
authority (variable number of resource records)		
additional information (variable number of resource records)		

IFT585 Télématique (T10-Application_v093a) 57

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Protocole DNS, messages

identification	flags	↑ 12 bytes ↓
number of questions	number of answer RRs	
number of authority RRs	number of additional RRs	
questions (variable number of questions)		
answers (variable number of resource records)		
authority (variable number of resource records)		
additional information (variable number of resource records)		

Champs **name et type** d'une requête →

RRs en réponse à la requête →

Enregistrements pour les serveurs de référence →

Information additionnelle « utile » qui peut être utilisée →

IFT585 Télématique (T10-Application_v093a) 58

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Sommaire

- Principes généraux
 - clients et serveurs
 - exigences
- Protocoles courants
 - HTTP, URL
 - DNS
 - **Telnet**
 - FTP
 - SMTP, POP3, IMAP, MIME
 - SSH, SSL, S/MIME
 - RTP, SIP
 - ...
- Serveur Web
 - Conception
 - Programmation
- Diffusion de contenu
 - « Web caching »
 - Réseaux de diffusion de contenu
 - Partage de fichiers P2P

IFT585 Télématique (T10-Application_v093a) 59

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Telnet

- Revenons en arrière, en 1974 (pas en 1969) !
- Vous êtes au commande du projet TCP/IP.
- Quels protocoles applicatifs de base proposeriez-vous ?
- Dans quel ordre les réaliseriez-vous ?

IFT585 Télématique (T10-Application_v093a) 60

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

Telnet

- La base de plusieurs autres protocoles dont FTP, POP, SMTP.
- RFC 854, 855
- Caractéristiques :
 - Paritaire
 - Échange de flux d'octets
 - US ASCII (7 bits – 128 caractères)

IFT585 Télématique (T10-Application_v093a) 61

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

Telnet

- Principe du NVT
- Séquences de caractères pouvant comprendre des commandes
- Configuration
 - négociation (1976)
 - DO
 - DON'T
 - WILL
 - WON'T
 - sub-négociation (1983)

IFT585 Télématique (T10-Application_v093a) 62

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Telnet

NAME	CODE	MEANING
SE	240	End of subnegotiation parameters.
NOP	241	No operation.
Data Mark	242	The data stream portion of a Synch. This should always be accompanied by a TCP Urgent notification.
Break	243	NVT character BRK.
Interrupt Process	244	The function IP.
Abort output	245	The function AO.
Are You There	246	The function AYT.
Erase character	247	The function EC.
Erase Line	248	The function EL.
Go ahead	249	The GA signal.
SB	250	Indicates that what follows is subnegotiation of the indicated option.
WILL (option code)	251	Indicates the desire to begin performing, or confirmation that you are now performing, the indicated option.
WON'T (option code)	252	Indicates the refusal to perform, or continue performing, the indicated option.
DO (option code)	253	Indicates the request that the other party perform, or confirmation that you are expecting the other party to perform, the indicated option.
DON'T (option code)	254	Indicates the demand that the other party stop performing, or confirmation that you are no longer expecting the other party to perform, the indicated option.
IAC	255	Data Byte 255.

IFT585 Télématique (T10-Application_v093a) 63

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Sommaire

<input type="checkbox"/> Principes généraux	<input type="checkbox"/> Serveur Web
■ clients et serveurs	■ Conception
■ exigences	■ Programmation
<input type="checkbox"/> Protocoles courants	<input type="checkbox"/> Diffusion de contenu
■ HTTP, URL	■ « Web caching »
■ DNS	■ Réseaux de diffusion de contenu
■ Telnet	■ Partage de fichiers P2P
■ FTP	
■ SMTP, POP3, IMAP, MIME	
■ SSH, SSL, S/MIME	
■ RTP, SIP	
■ ...	

IFT585 Télématique (T10-Application_v093a) 64

UNIVERSITÉ DE SHERBROOKE 2009-03-28

FTP : Le transfert de fichiers

Utilisateur sur l'équipement

IPM FTP client FTP

système de fichiers local

transfert de fichiers

Serveur FTP

système de fichiers à distance

- transfert de fichiers de/vers l'équipement distant
- modèle client/serveur
 - client : démarre le transfert (de/vers l'équipement distant)
 - serveur : équipement distant
- ftp: RFC 959
- serveur ftp : port 21

IFT585 Télématique (T10-Application_v093a) 65

UNIVERSITÉ DE SHERBROOKE 2009-03-28

FTP : Deux connexions : commande et transfert

Client FTP

gestion de la connexion TCP - port 21

communication de données TCP - port 20

Server FTP

- Le protocole utilise deux connexions : une pour la commande et l'autre pour le transfert des fichiers
- Dans les deux cas, TCP est le transporteur.
- La connexion de commande est ouverte par le client, celle de transfert par le serveur sur un port du client dont le numéro a été transmis via la connexion de commande (d'où un problème avec NAT).
- Après le transfert du fichier, le serveur ferme la connexion de transfert.
- Le serveur ouvre une nouvelle connexion pour chaque transfert subséquent.
- FTP est un protocole à états (par exemple : dossier courant, information d'authentification, etc.)

IFT585 Télématique (T10-Application_v093a) 66

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Commandes FTP, réponses

Exemples de commandes

- commandes envoyées au format ASCII sur le canal de transmission
 - **USER *username***
 - **PASS *password***
 - **LIST** : retourne la liste des fichiers du dossier courant
 - **RETR *filename*** : extrait le fichier
 - **STOR *filename*** : stocke le fichier sur l'équipement distant

Exemples de codes de retour

- code d'état et phrase (comme dans HTTP)
 - **331 Username OK, password required**
 - **125 data connection already open; transfer starting**
 - **425 Can't open data connection**
 - **452 Error writing file**

IFT585 Télématique (T10-Application_v093a) 67

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Sommaire

- Principes généraux
 - clients et serveurs
 - exigences
- Protocoles courants
 - HTTP, URL
 - DNS
 - Telnet
 - FTP
 - **SMTP, POP3, IMAP, MIME**
 - SSH, SSL, S/MIME
 - RTP, SIP
 - ...

- Serveur Web
 - Conception
 - Programmation
- Diffusion de contenu
 - « Web caching »
 - Réseaux de diffusion de contenu
 - Partage de fichiers P2P

IFT585 Télématique (T10-Application_v093a) 68

UNIVERSITÉ DE SHERBROOKE 2009-03-28

SMTP

Les principaux composants

3 principaux composants

- ❑ agents
- ❑ serveurs de messagerie
- ❑ SMTP : Simple Mail Transfer Protocol

Agent

- ❑ alias « mail reader »
- ❑ saisie, édition, lecture des courriels
- ❑ Eudora, Outlook, Mail

Serveur

- ❑ stocke les messages entrants et sortants
- ❑ achemine les messages sortants
- ❑ transfère les messages entrants

IFT585 Télématique (T10-Application_v093a) 69

UNIVERSITÉ DE SHERBROOKE 2009-03-28

SMTP

Les services du serveur de messagerie

Serveurs de messagerie

- ❑ **boîte de réception** contient les messages entrants de l'utilisateur
- ❑ **file d'attente** contient les courriels à envoyer
- ❑ **protocole SMTP** utilisé par les serveurs de messagerie pour l'échange de messages
 - client : serveur d'envoi des messages
 - « serveur » : serveur de réception des messages

IFT585 Télématique (T10-Application_v093a) 70

UNIVERSITÉ DE SHERBROOKE 2009-03-28

SMTP [RFC 2821]

- utilise TCP pour transférer de façon fiable les courriels du client au serveur, port 25
- transfert direct : du serveur d'envoi au serveur de réception
- trois étapes de transfert
 - établissement de la liaison (handshaking)
 - transfert de messages
 - fermeture
- interaction commande/réponse
 - **commande** : texte ASCII
 - **réponse** : code et description du statut
- les messages doivent être au format ASCII 7-bit

IFT585 Télématique (T10-Application_v093a) 71

UNIVERSITÉ DE SHERBROOKE 2009-03-28

SMTP Un cas : Sylvie envoie un message à Luc

1. Sylvie utilise un agent pour rédiger le message et inscrit l'adresse courriel de Luc dans le champ « to » : `luc@www.uneEcoleX.org`
2. L'agent de Sylvie envoie le message au serveur de messagerie par le protocole SMTP.
3. Le message est ajouté dans la file d'attente des messages sur le serveur.
4. Le serveur (de Sylvie) transfère le message au serveur de Luc en utilisant aussi le protocole SMTP.
5. Le serveur de messagerie de Luc dépose le message dans sa boîte de réception.
6. Luc lit le message à l'aide de son agent grâce au protocole POP3 ou IMAP.

IFT585 Télématique (T10-Application_v093a) 72

UNIVERSITÉ DE SHERBROOKE 2009-03-28

SMTP - Exemple de dialogue suite à l'ouverture d'une connexion par un client

```

S: 220 uneEcoleX.org
C: HELO ecoles.qc
S: 250 Hello ecoles.qc, pleased to meet you
C: MAIL FROM: <sylvie@ecoles.qc>
S: 250 sylvie@ecoles.qc... Sender ok
C: RCPT TO: <luc@uneEcoleX.org>
S: 250 luc@uneEcoleX.org ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C: Bonjour Luc,
C: Voici la photo que tu voulais.
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 uneEcoleX.org closing connection

```

IFT585 Télématique (T10-Application_v093a) 73

UNIVERSITÉ DE SHERBROOKE 2009-03-28

SMTP

Un exercice

- telnet smtp.USherbrooke.ca 25
- Consulter la réponse 220 du serveur
- Le fait de saisir les commandes HELO, MAIL FROM, RCPT TO, DATA, QUIT permet d'envoyer des courriels sans agent du côté client

IFT585 Télématique (T10-Application_v093a) 74

UNIVERSITÉ DE SHERBROOKE 2009-03-28

SMTP Conclusion

- ❑ SMTP utilise des connexions persistantes
- ❑ SMTP requiert que le message (entête et corps) soit au format ASCII 7-bit
- ❑ Serveur SMTP utilise « CRLF.CRLF » pour déterminer la fin d'un message.
- ❑ Il manque quelque chose... la récupération du message par le destinataire!

SMTP versus HTTP

- ❑ **HTTP** : pull (principalement)
SMTP : push
- ❑ **Les deux** : interaction commande/réponse au format ASCII, codes de statut
- ❑ **HTTP** : chaque objet est encapsulé dans son propre message réponse.
SMTP : plusieurs objets envoyés dans un message à sections multiples.

IFT585 Télématique (T10-Application_v093a) 75

UNIVERSITÉ DE SHERBROOKE 2009-03-28

SMTP Format des messages textuels

- ❑ Voir RFC 822
- ❑ lignes d'entête
 - À/To:
 - De/From:
 - Sujet/Subject:
 - etc.
 - (autres options SMTP)
- ❑ corps
 - le « message », caractères ASCII seulement

IFT585 Télématique (T10-Application_v093a) 76

UNIVERSITÉ DE SHERBROOKE 2009-03-28

MIME

Format des messages « multimédia »

- ❑ MIME (multimédia mail extension) : RFC 2045, 2056
- ❑ Des lignes supplémentaires sont ajoutées dans l'entête pour spécifier que le contenu est de type MIME

version de MIME

méthode d'encodage des données

paramètres de déclaration du type/sous-type des données multimédia

données encodées

```

From: sylvie@ecoles.qc
To: luc@uneEcoleX.org
Subject: Photo.
MIME-Version: 1.0
Content-Transfer-Encoding: base64
Content-Type: image/jpeg

base64 encoded data .....
.....base64 encoded data

```

IFT585 Télématique (T10-Application_v093a) 77

UNIVERSITÉ DE SHERBROOKE 2009-03-28

MIME

Content-Type: type/sous-type; paramètres

- Texte**
 - sous-types : plain, html
- Image**
 - sous-types : jpeg, gif
- Audio**
 - sous-types :
 - basic (8-bit mu-law encoded)
 - 32kadpcm (32 kbps coding)
- Vidéo**
 - sous-types : mpeg, quicktime
- Application**
 - Les autres données qui doivent être traitées par le lecteur pour les rendre « consultables »
 - sous-types : msword, octet-stream

IFT585 Télématique (T10-Application_v093a) 78

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

MIME Multipart

```

From: sylvie@ecoles.qc
To: luc@uneEcoleX.org
Subject: Photo.
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary=StartOfNextPart

--StartOfNextPart
Bonjour Luc, voici la photo que je t'avais promise.
--StartOfNextPart
Content-Transfer-Encoding: base64
Content-Type: image/jpeg
base64 encoded data .....
.....base64 encoded data
--StartOfNextPart
Elle est belle non ?

```

IFT585 Télématique (T10-Application_v093a) 79

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

Protocoles d'accès Mise en situation

```

graph LR
    A[agent] -- SMTP --> B[serveur de l'émetteur]
    B -- SMTP --> C[serveur du récepteur]
    C -- "Protocole d'accès" --> D[agent]

```

- SMTP: livre/stocke dans le serveur du récepteur
- Protocoles d'accès (« Mail access protocol »)
 - Extraction du serveur
 - Exemples
 - POP, POP3: Post Office Protocol [RFC 1939]
 - autorisation (agent <--> serveur) et téléchargement
 - IMAP: Internet Mail Access Protocol [RFC 1730]
 - plus d'options (plus complexe)
 - gestion des messages stockés sur le serveur

IFT585 Télématique (T10-Application_v093a) 80

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Protocoles d'accès Accès HTTP à la « hotmail »

Un même agent donne accès à plusieurs services offerts par des protocoles divers.

Le serveur applicatif est un agent délégué.

IFT585 Télématique (T10-Application_v093a) 81

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Protocoles d'accès POP3 et IMAP

POP3

- Les exemples précédents utilisent le mode «télécharger et supprimer»
- Si Luc change de client, il ne peut pas lire les messages déjà lus.
- «télécharger et supprimer»: messages sur différents clients
- POP3 ne conserve pas l'état d'une session à l'autre

IMAP

- Tous les messages sont conservés à un seul endroit: le serveur.
- Permet à l'utilisateur d'organiser ses messages en dossiers.
- IMAP conserve l'état d'une session à l'autre :
 - noms des dossiers et correspondance entre l'ID d'un message et le nom du dossier

IFT585 Télématique (T10-Application_v093a) 82

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Protocoles d'accès Un exemple avec POP3

Phase d'autorisation

- Commandes du client
 - **user**: nom utilisateur
 - **pass**: mot de passe
- Réponses du serveur
 - **+OK**
 - **-ERR**

Phase de transaction

- Commandes du client
 - **list**: liste les numéros de message
 - **retr**: extrait les messages par numéro
 - **dele**: supprime
 - **quit**

```

S: +OK POP3 Serveur ready
C: user luc
S: +OK
C: pass hungry
S: +OK user successfully logged on

C: list
S: 2 912
S: .
C: retr 1
S: <message 1 contents>
S: .
C: dele 1
C: retr 2
S: <message 1 contents>
S: .
C: dele 2
C: quit
S: +OK POP3 Serveur signing off

```

IFT585 Télématique (T10-Application_v093a) 83

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Sommaire

- Principes généraux
 - clients et serveurs
 - exigences
- Protocoles courants
 - HTTP, URL
 - DNS
 - Telnet
 - FTP
 - SMTP, POP3, IMAP, MIME
 - SSH, SSL, S/MIME
 - RTP, SIP
 - ...
- **Serveur Web**
 - **Conception**
 - **Programmation**
- Diffusion de contenu
 - « Web caching »
 - Réseaux de diffusion de contenu
 - Partage de fichiers P2P

IFT585 Télématique (T10-Application_v093a) 84

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Créer un serveur Web simple

Traitement une requête HTTP

- Recevoir la requête
- Analyser l'entête
- Obtenir le fichier demandé du serveur de fichiers
- Construire le message de réponse HTTP
 - en-tête
 - charge utile (fichier)
- Envoyer la réponse au client

IFT585 Télématique (T10-Application_v093a) 85

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Sommaire

- Principes généraux
 - clients et serveurs
 - exigences
- Protocoles courants
 - HTTP, URL
 - DNS
 - Telnet
 - FTP
 - SMTP, POP3, IMAP, MIME
 - SSH, SSL, S/MIME
 - RTP, SIP
 - ...
- Serveur Web
 - Conception
 - Programmation
- Diffusion de contenu
 - « Web caching »
 - Réseaux de diffusion de contenu
 - Partage de fichiers P2P

IFT585 Télématique (T10-Application_v093a) 86

UNIVERSITÉ DE SHERBROOKE 2009-03-28

« Web caching » (serveur proxy)

Objectif : répondre à la requête du client sans impliquer le serveur d'origine

- configuration du fureteur de l'utilisateur = accès au Web via cache
- le fureteur envoie toutes les requêtes HTTP au cache :
 - objet en cache: le cache retourne l'objet
 - sinon le cache demande l'objet au serveur d'origine, puis retourne l'objet au client

IFT585 Télématique (T10-Application_v093a) 87

UNIVERSITÉ DE SHERBROOKE 2009-03-28

« Web caching » Comment et pourquoi

- joue le rôle de client et de serveur
- peut vérifier si la version de l'objet est à jour à l'aide de If-modified-since de l'entête HTTP
 - Question : le cache doit-il prendre le risque et livrer les objets du cache sans vérification ?
 - Utilisation d'heuristiques.
- est typiquement installé par les FAI

Pourquoi le Web caching ?

- réduit le temps de réponse des requêtes du client
- réduit le trafic sur un lien d'accès institutionnel
- permet à certains fournisseurs de contenu d'effectivement livrer le contenu.

IFT585 Télématique (T10-Application_v093a) 88

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Exemple de caching (1)

Hypothèses

- ❑ taille moyenne des objets = 100 000 bits
- ❑ taux moyen des requêtes = 15 / s
- ❑ délai de l'aiguilleur institutionnel à n'importe quel serveur d'origine et retour à l'aiguilleur = 2 s
- ❑ délai sur le LAN < 1 ms

Conséquences

- ❑ utilisation du LAN = 15%
- ❑ utilisation de liaison d'accès = 100%
- ❑ la saturation de la liaison d'accès entraîne un délai pouvant atteindre plusieurs secondes, voire quelques minutes
- ❑ délai total = délai Internet + délai accès + délai LAN > 2 s + 1 s + 1 ms > 3 s (peut atteindre quelques minutes)

serveurs d'origine

Internet public

liaison d'accès 1.5 Mbps

réseau institutionnel

10 Mbps LAN

cache institutionnel

IFT585 Télématique (T10-Application_v093a) 89

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Exemple de caching (2)

Solution possible

- ❑ augmenter la largeur de bande de la liaison d'accès à, disons, 10 Mbps

Conséquences

- ❑ utilisation du LAN = 15%
- ❑ utilisation de liaison d'accès = 15%
- ❑ délai total = délai Internet + délai accès + délai LAN ~ 2 s + 1 ms + 1 ms ~ 2,002 s
- ❑ souvent une mise à niveau coûteuse

serveurs d'origine

Internet public

liaison d'accès 10 Mbps

réseau institutionnel

10 Mbps LAN

cache institutionnel

IFT585 Télématique (T10-Application_v093a) 90

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Exemple de caching (3)

Cache installé

- ❑ on suppose un taux de réussite de 40%

Conséquences

- ❑ 40% des requêtes seront satisfaites presque immédiatement
- ❑ 60% des requêtes sont satisfaites par le serveur d'origine
- ❑ utilisation de la liaison d'accès réduite à 60%, résultant en des délais négligeables (disons 20 ms)
- ❑ délai total
= délai Internet + délai accès + délai LAN
= $0,6 \cdot 2 \text{ s} + 0,6 \cdot 2 \text{ ms} + 1 \text{ ms}$
< 1,5 s

IFT585 Télématique (T10-Application_v093a) 91

UNIVERSITÉ DE SHERBROOKE 2009-03-28

Réseau de distribution de contenu (Content distribution network, CDN)

- ❑ Les fournisseurs de contenu sont les clients utilisateurs des réseaux CDN.

Reproduction du contenu

- ❑ Les sociétés CDN installent des centaines de serveurs CDN à travers l'Internet
 - ❑ FAI de bas niveau -> près des utilisateurs
- ❑ CDN reproduit le contenu de ses clients sur des serveurs CDN. Lorsque le fournisseur met à jour le contenu, CDN met à jour les serveurs

IFT585 Télématique (T10-Application_v093a) 92

UNIVERSITÉ DE SHERBROOKE 2009-03-28

CDN Exemple

① HTTP demande
www.foo.com/sports/ruth.html
Serveur d'origine

② DNS recherche www.cdn.com
Serveur DNS de référence

③ HTTP demande
www.cdn.com/www.foo.com/sports/ruth.gif
près serveur CDN

Serveur d'origine

- www.foo.com
- distribue le fichier HTML
- remplace `http://www.foo.com/sports.ruth.gif` par `http://www.cdn.com/www.foo.com/sports/ruth.gif`

Compagnie CDN

- cdn.com
- distribue le fichier gif
- utilise son serveur DNS de référence pour aiguiller et rediriger les requêtes

IFT585 Télématique (T10-Application_v093a) 93

UNIVERSITÉ DE SHERBROOKE 2009-03-28

CDN Pour en savoir un savoir un peu plus

aiguillage de requêtes

- CDN crée une « carte » qui indique la distance entre les feuilles FAI et les nœuds CDN
- lorsqu'une requête arrive au serveur DNS de référence :
 - le serveur détermine le FAI d'où provient la requête
 - utilise la « carte » pour déterminer le meilleur serveur CDN

autres applications

- lecture en continu d'audio et vidéo
- lecture en temps-réel d'audio et vidéo

généralisation

- les nœuds CDN créent un arbre de recouvrement minimal pouvant être ensuite étendu à une « structure » de recouvrement partiellement redondante

IFT585 Télématique (T10-Application_v093a) 94

UNIVERSITÉ DE SHERBROOKE 2009-03-28

P2P

Poste à poste, entre pairs, « peer to peer »

Exemple :

1. Sylvie exécute un client P2P sur son poste.
2. Elle se connecte à l'Internet de façon intermittente et obtient une nouvelle adresse IP à chaque nouvelle connexion.
3. Elle recherche la chanson « Hey Jude ».
4. L'application affiche la liste des pairs qui ont une copie de « Hey Jude » sur leur poste.
5. Sylvie choisit un pair : Luc.
6. Le fichier est copié du poste de Luc vers le poste de Sylvie.
7. Pendant que Sylvie télécharge à partir du poste de Luc, d'autres pairs téléchargent d'autres fichiers à partir de son poste.
8. Sylvie est un pair qui est **en même temps un client et un serveur**.

IFT585 Télématique (T10-Application_v093a) 95

UNIVERSITÉ DE SHERBROOKE 2009-03-28

P2P

Pourquoi tout ce tapage ?

<ul style="list-style-type: none"> <input type="checkbox"/> Le P2P représente le fonctionnement de l'Internet tel qu'il a été conçu et construit. <input type="checkbox"/> En définitive, l'expression ne devrait même pas exister. 	<ul style="list-style-type: none"> <input type="checkbox"/> Pénurie d'adresses <ul style="list-style-type: none"> <input type="checkbox"/> NAT <input type="checkbox"/> Sécurité <ul style="list-style-type: none"> <input type="checkbox"/> Coupe-feu <input type="checkbox"/> Droit d'auteur <ul style="list-style-type: none"> <input type="checkbox"/> Contournement des canaux de distribution <input type="checkbox"/> Sureté <ul style="list-style-type: none"> <input type="checkbox"/> Crime organisé <input type="checkbox"/> Terrorisme
---	---

IFT585 Télématique (T10-Application_v093a) 96

UNIVERSITÉ DE SHERBROOKE 2009-03-28

P2P Répertoire centralisé

Conception originale de « Napster »

- Lorsqu'un pair se connecte, il envoie au serveur central :
 - son adresse IP,
 - son contenu.
- Sylvie cherche « Hey Jude ».
- Sylvie demande le fichier à Luc.

IFT585 Télématique (T10-Application_v093a) 97

UNIVERSITÉ DE SHERBROOKE 2009-03-28

P2P Problèmes découlant du répertoire centralisé

- Point de défaillance unique
- Goulot d'étranglement (*performance bottleneck*)

Le transfert de fichiers est décentralisé,
mais la localisation d'un contenu est hautement centralisée

IFT585 Télématique (T10-Application_v093a) 98

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

P2P

Répertoire décentralisé

- Chaque pair est soit chef de groupe ou est membre du groupe d'un chef de groupe.
- Le chef de groupe garde la trace du contenu de tous les membres de son groupe.
- Les pairs envoient leurs demandes au chef de groupe ; seul le chef de groupe peut envoyer des demandes aux autres chefs de groupe.

● pair
● « group leader »
— structure de rouvrement

IFT585 Télématique (T10-Application_v093a) 99

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

P2P

En savoir un peu plus sur les répertoires décentralisés

réseau dédié

- pairs sont des nœuds
- en périphérie : pairs et leur chef de groupe
- en périphérie : pairs de chefs de groupe
- « voisins » virtuels

nœud d'amorce (bootstrap node)

- lorsqu'un pair se connecte, il est assigné à un chef de groupe ou est nommé chef de groupe.

avantages de l'approche

- pas de répertoire centralisé sur un serveur
 - service de localisation distribué entre les pairs
 - plus difficile à mettre hors service

désavantages de l'approche

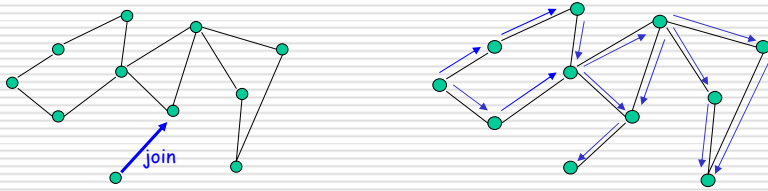
- nœud d'amorce nécessaire
- les chefs de groupe peuvent être surchargés

IFT585 Télématique (T10-Application_v093a) 100

UNIVERSITÉ DE SHERBROOKE 2009-03-28

P2P Recherche par inondation – les principes

- Pas de hiérarchie
- Utilisation d'un « nœud d'amorce » pour découvrir les autres
- Message d'abonnement (« join »)
- Envoie une demande aux voisins
- Voisins transfèrent la demande
- Si un pair qui a reçu la demande a l'objet, il envoie un message au pair demandeur



IFT585 Télématique (T10-Application_v093a) 101

UNIVERSITÉ DE SHERBROOKE 2009-03-28

P2P Recherche par inondation - Analyse

<p><u>Pour</u></p> <ul style="list-style-type: none"> <input type="checkbox"/> les responsabilités des pairs sont similaires : il n'y a pas de chef de groupe <input type="checkbox"/> hautement décentralisé <input type="checkbox"/> aucun pair ne conserve de répertoire d'information 	<p><u>Contre</u></p> <ul style="list-style-type: none"> <input type="checkbox"/> requête entraîne trafic excessif <input type="checkbox"/> rayon de recherche : contenu peut ne pas être présent <input type="checkbox"/> nœud d'amorce <input type="checkbox"/> maintenance du réseau de recouvrement
--	--

Un exemple : Gnutella

IFT585 Télématique (T10-Application_v093a) 102

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

P2P et le droit d'auteur

- Mise en contexte
 - imprimerie, photocopie, magnétophone, cassette, ...
- Usages légitimes
 - antécédents
 - équilibre
- Arsenal
 - préventif
 - correctif
 - coercitif

IFT585 Télématique (T10-Application_v093a) 103

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

P2P à la recherche de la légitimité

- Skype, iPlayer, Joost
- Mouvance P4P
- Références
 - http://techno.branchez-vous.com/actualite/2008/01/le_posteaposte_pas_seulement_p.html
 - http://www.news.com/Harnessing-the-power-of-P2P/2100-1034_3-6227406.html?part=rss&tag=2547-1_3-0-5&subj=news

IFT585 Télématique (T10-Application_v093a) 104

UNIVERSITÉ DE
SHERBROOKE

2009-03-28

Synthèse

À propos des protocoles d'application

- échange typique de messages requête/réponse:
 - client demande info ou service
 - serveur répond avec données, code de statut
- formats du message:
 - entête : champs qui fournissent des infos sur les données
 - données : infos transmises
- contrôle
 - intégré aux messages
 - externe aux messages
- centralisé ou non
- avec ou sans état
- transfert fiable ou non
- sécurité
 - authentification
 - confidentialité
 - ...

IFT585 Télématique (T10-Application_v093a) 105