

Partie 2 : Applications de l'Internet de type Client/Serveur

Olivier GLÜCK
Université LYON 1/Département Informatique
Olivier.Gluck@univ-lyon1.fr
<http://perso.univ-lyon1.fr/olivier.gluck>



1

Copyright

- Copyright © 2023 Olivier Glück; all rights reserved
- Ce support de cours est soumis aux droits d'auteur et n'est donc pas dans le domaine public. Sa reproduction est cependant autorisée à condition de respecter les conditions suivantes :
 - Si ce document est reproduit pour les besoins personnels du reproducteur, toute forme de reproduction (totale ou partielle) est autorisée à la condition de citer l'auteur.
 - Si ce document est reproduit dans le but d'être distribué à des tierces personnes, il devra être reproduit dans son intégralité sans aucune modification. Cette notice de copyright devra donc être présente. De plus, il ne devra pas être vendu.
 - Cependant, dans le seul cas d'un enseignement gratuit, une participation aux frais de reproduction pourra être demandée, mais elle ne pourra être supérieure au prix du papier et de l'encre composant le document.
 - Toute reproduction sortant du cadre précisé ci-dessus est interdite sans accord préalable écrit de l'auteur.

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

2

2

Remerciements

- Certains transparents sont basés sur des supports de cours de :
 - Olivier Aubert (LYON 1)
 - Olivier Fourmaux (UPMC)
 - Bénédicte Le Grand (UPMC)
- Des figures sont issues des livres cités en bibliographie

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

3

3

Bibliographie

- « *Réseaux* », 4ième édition, Andrew Tanenbaum, Pearson Education, ISBN 2-7440-7001-7
- « *La communication sous Unix* », 2ième édition, Jean-Marie Rifflet, Ediscience international, ISBN 2-84074-106-7
- « *Analyse structurée des réseaux* », 2ième édition, J. Kurose et K. Ross, Pearson Education, ISBN 2-7440-7000-9
- « *TCP/IP Illustrated Volume 1, The Protocols* », W. R. Stevens, Addison Wesley, ISBN 0-201-63346-9
- « *TCP/IP Architecture, protocoles, applications* », 4ième édition, D. Comer, Dunod, ISBN 2-10-008181-0
- Internet...
 - <http://www.w3.org/>
 - <http://www.rfc-editor.org/> (documents normatifs dans TCP/IP)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

4

4

Plan de la partie 2

- Introduction / Rappel
- Connexions à distance (telnet/rlogin/rsh/ssh/X11)
- Applications de transfert de fichiers (FTP/TFTP)
- Accès aux fichiers distants (NFS/SMB)
- Gestion d'utilisateurs distants (NIS)
- DNS : un annuaire distribué
- LDAP : un annuaire fédérateur sécurisé
- La messagerie électronique (SMTP/POP/IMAP)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

5

5

Introduction / Rappels



6

La couche application

- La couche application
 - gère les logiciels utilisateurs (applications) en s'appuyant sur les services de bout en bout définis dans les couches de niveau inférieur
 - repose généralement sur le modèle Client/Serveur (modèle requête/réponse)
 - supporte les environnements hétérogènes
- On distingue l'application et le protocole applicatif
 - le protocole applicatif définit les échanges entre les parties cliente et serveur de l'application
 - une interface (API) permet au protocole applicatif d'utiliser les services de bout-en-bout fournis par un protocole de transport sous-jacent

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

7

Quel service de transport ?

- Socket = interface entre le processus applicatif et le protocole de transport
 - Côté émetteur : l'application envoie des messages par la porte
 - De l'autre côté de la porte, le protocole de transport doit déplacer les messages à travers le réseau, jusqu'à la porte du processus receveur
- De nombreux réseaux (dont Internet) fournissent plusieurs protocoles de transport
 - Lequel choisir lorsqu'on développe une application ?
 - Étude des services fournis par chaque protocole
 - Sélection du protocole qui correspond le mieux aux besoins de l'application

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

8

Quel service de transport ?

- Faut-il choisir le train ou l'avion pour faire Paris/Nice ?
 - tout dépend des critères du voyageur (rapidité, confort, sécurité, prix, arrivée en centre ville...)
- 3 types de besoins au niveau des applications :
 - fiabilité du transfert (S'autorise t-on à perdre quelques données ? Dans quelle proportion ?)
 - bande passante (Quelle est la taille minimale du tuyau de communication ?)
 - délai : latence et gigue (variation du délai)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

9

Quel service de transport ?

- Fiabilité du transfert
 - Certaines applications nécessitent une fiabilité à 100%
 - Courrier électronique (SMTP)
 - Transfert de fichiers (FTP)
 - Accès distant (Telnet)
 - Transfert de documents Web (HTTP)
 - Applications financières
 - D'autres peuvent tolérer des pertes (loss-tolerant applications)
 - Applications multimédia : audio/vidéo (une perte d'une faible quantité de données n'induit qu'une petite irrégularité dans l'écoute ou la vision du film)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

10

Quel service de transport ?

- Bande passante
 - Certaines applications requièrent une bande passante minimale
 - Téléphonie sur Internet : si la voix est codée à 32 Kbps, les données doivent être transmises et reçues à ce débit
 - Applications multimédia
 - D'autres utilisent la bande passante disponible (applications élastiques)
 - Courrier électronique, transfert de fichiers, accès distant, Web
 - Plus il y a de bande passante, mieux c'est !

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

11

Quel service de transport ?

- Délai (contraintes temporelles)
 - Certaines applications nécessitent un délai de bout-en-bout faible (moins de quelques centaines de ms)
 - Applications temps réel interactives :
 - Téléphonie sur Internet
 - Environnements virtuels
 - Téléconférence
 - Jeux en réseau
 - Pour les applications non temps réel, un délai court est préférable, mais pas de contrainte forte

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

12

7

8

9

10

11

12

Quel service de transport ?

Application	Pertes	Bande passante	Sensibilité temp.
Transfert de fichiers	sans perte	élastique	Non
e-mail	sans perte	élastique	Non
Pages Web	sans perte	élastique	Non
Audio/vidéo temps réel	tolérant	audio: 5Kb - 1Mb vidéo: 10Kb - 5Mb	Oui, 100's ms
Audio/vidéo enregistré	tolérant	idem	Oui, quelques s
Jeux interactifs	tolérant	quelques Kbps	Oui, 100's ms
Applis financières	sans perte	élastique	Oui et non

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

13

13

Services proposés dans Internet

Service TCP :

- **orienté connexion** : connexion nécessaire entre le client et le serveur
- **transport fiable** entre le processus émetteur et récepteur
- **contrôle de flot** : l'émetteur ne submerge pas le récepteur
- **contrôle de congestion** : réduit le débit de l'émetteur quand le réseau est congestionné
- ne propose pas :
 - de garantie de délai,
 - de bande passante minimale

Service UDP :

- transfert de données non fiable
- ne propose pas
 - de connexion,
 - de fiabilité,
 - de contrôle de flot,
 - de contrôle de congestion,
 - de garantie temporelle,
 - de bande passante
- beaucoup plus simple que TCP (UDP=IP) donc plus rapide
- pas de limitation du débit

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

14

14

Principales applications Internet

Application	Protocole applicatif	Protocole de transport
e-mail	SMTP [RFC 821,2821]	TCP
Accès distant	telnet [RFC 854]	TCP
Web	HTTP [RFC 2068,2616]	TCP
Transfert de fichiers	FTP [RFC 959]	TCP
Streaming multimedia	propriétaire	TCP ou UDP
Serveur Fichiers	NFS	TCP ou UDP
Voix sur IP	propriétaire	En général UDP

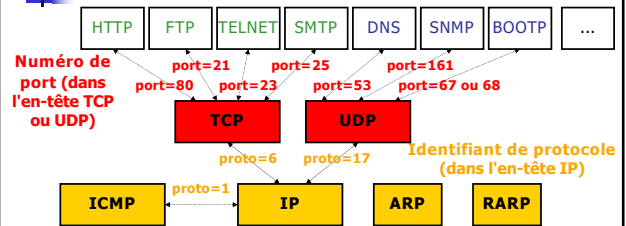
Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

15

15

Principales applications Internet



Tous les ports sur <http://www.iana.org/assignments/port-numbers>

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

16

16

Applications de connexion à distance

Principes
telnet, rlogin, rsh, ssh, X11



17

Connexions à distance

- Application permettant à un utilisateur de se connecter (prendre partiellement le contrôle)
 - sur un ordinateur distant (à partir d'un terminal local)
 - pourvu que cet utilisateur dispose d'un accès autorisé à cette machine
 - exécution de commandes saisies localement au clavier sur une machine distante
 - les environnements local et distant peuvent être hétérogènes (windows-->unix, ...)

Émulation de terminaux :



Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

18

18

Connexions à distance

- Plusieurs protocoles
 - telnet : le standard (existe sur de nombreuses plate-formes)
 - rlogin : uniquement entre machines unix
 - ssh : sécurisé (authentification + cryptage), peut transporter le DISPLAY
- Besoin de l'application : inter-activité
 - tout ce qui est tapé au clavier sur le client est envoyé sur la connexion au serveur
 - tout ce qui est envoyé par le serveur vers le client, sur la connexion, est affiché dans le terminal

Olivier Glück

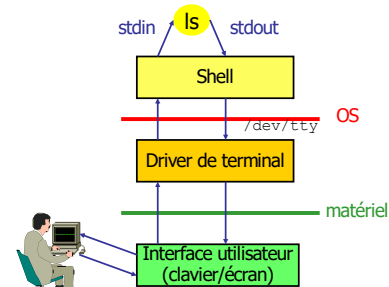
M2 SRS - Admin Systèmes et Réseaux

19

19

Connexions locales

- Fonctionnement d'une connexion locale



Olivier Glück

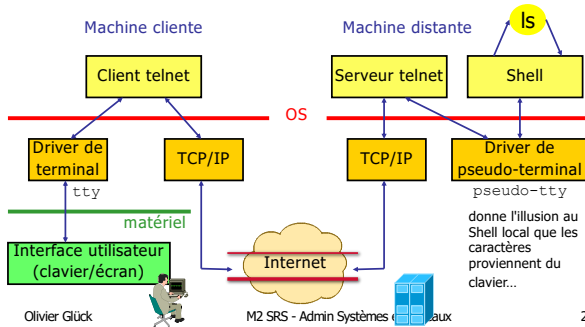
M2 SRS - Admin Systèmes et Réseaux

20

20

Connexions distantes

- Fonctionnement d'une connexion distante



Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

21

21

Telnet : protocole et application

TELEcommunication NETWORK protocol

- un des premiers standard de l'Internet : RFC 854,855 (1983)
- connexion TCP sur le port 23 côté serveur
- authentification sur le shell distant (mot de passe en clair)
- quand un caractère est tapé au clavier, il est envoyé au serveur qui renvoie un "écho" du caractère ce qui provoque son affichage dans le terminal local
- prise en compte de l'hétérogénéité
 - mécanisme de négociation d'options à la connexion (codage des caractères ASCII sur 7 ou 8 bits ?)
 - exemple : telnet d'une machine Windows vers une machine Unix --> tous les caractères ASCII n'ont pas la même signification

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

22

22

Exécution de Telnet

- Les différentes exécutions possibles (côté client)
 - sans argument (paramétrer sa connexion distante)
telnet
 - par le nom de la machine distante (DNS+port 23)
telnet nom_du_serveur
 - par l'adresse IP de la machine distante (port 23)
telnet adr_IP_du_serveur
 - accès à un autre service (connexion sur un autre port)
telnet adr_IP_du_serveur numéro_port

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

23

23

Exécution de Telnet

```

lima
ogluck@lima:~$ telnet www.ens-lyon.fr 80
Trying 140.77.167.3...
Connected to pingouin.ens-lyon.fr.
Escape character is '^]'.
HEAD /index.html HTTP/1.0

HTTP/1.1 200 OK
Date: Mon, 01 Mar 2004 18:37:42 GMT
Server: Apache/1.3.26 (Unix) Debian GNU/Linux PHP/4.1.2
Last-Modified: Tue, 23 Oct 2001 08:59:37 GMT
ETag: "3fa02b-100e-3bd53179"
Accept-Ranges: bytes
Content-Length: 4110
Connection: close
Content-Type: text/html; charset=iso-8859-1

Connection closed by foreign host.
ogluck@lima:~$
    
```

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

24

24

Telnet : gestion de l'hétérogénéité

- Exemples d'hétérogénéité liée à l'OS :
 - interprétation du saut de ligne
 - sous UNIX : CR-LF
 - sous Windows : CR
 - interruption du processus en cours d'exécution
 - Ctrl-c sur certains systèmes, ESC sur d'autres...
- NVT - *Network Virtual Terminal*
 - un terminal "virtuel réseau" qui permet de supporter des environnements hétérogènes
 - réalise la conversion des caractères spéciaux ou des séquences particulières en un format NVT

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 25

25

Telnet : gestion de l'hétérogénéité

- Le format NVT
 - tous les caractères sont codés sur 8 bits
 - les 128 caractères ASCII sont transmis tels quels
 - NVT redéfinit la signification de certains caractères de commande ASCII :
 - CR (13_d) = retour au début de la ligne
 - LF (10_d) = déplacement d'une ligne vers le bas
 - > dans le format NVT, RETURN ou ENTER se traduit par CR-LF

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 26

26

Telnet : gestion de l'hétérogénéité

- Format NVT
 - le 8^{ème} bit à 1 permet de définir des caractères de commande NVT spécifiques (touches virtuelles)
 - chaque caractère de commande est précédé d'un caractère d'échappement spécifique : IAC

Commande	Code	Signification
IAC	255	L'octet suivant n'est pas une donnée mais un caractère de contrôle
DON'T	254	Rejette une demande d'exécution d'une option donnée (fais pas)
DO	253	Accepte une demande d'exécution d'une option donnée (fais...)
WON'T	252	Refus de mettre en œuvre une option donnée (je ne vais pas...)
WILL	251	Acceptation de mettre en œuvre une option donnée (je vais...)
SB	250	Début de négociation d'option
EL	248	Effacer la ligne
IP	244	Interruption du processus
SE	240	Fin de négociation d'option

La séquence Ctrl-c sous unix se traduira par la séquence NVT IAC IP

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 27

27

Telnet : gestion de l'hétérogénéité

- Envoi des commandes NVT hors-bande
 - > utilisation des données urgentes TCP
 - pourquoi ? Exemple :
 - le processus distant tourne en boucle, il ne lit/écrit plus aucune donnée sur le pseudo-terminal
 - les données en provenance ou à destination du client remplissent les tampons
 - le contrôle de flux TCP empêche le client et le serveur telnet de communiquer
 - le client ne peut même plus interrompre le processus fautif sur le serveur
 - > les données urgentes ne sont pas soumises au contrôle de flux TCP

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 28

28

Telnet : les options

- Exemples :
 - l'écho peut être fait par le site distant ou localement (dépend de la charge du réseau...)

- mode ligne : envoi ligne par ligne plutôt que caractère par caractère
- type du terminal (couleur, redimensionnement...), vitesse du terminal, taille de la fenêtre, ...

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 29

29

Telnet : négociation d'options

Send	Reply	Meaning
DO transmit binary	WILL transmit binary	
DO window size	WILL window size	Can we negotiate window size?
SB Window size 0 80 0 24 SE		Specify window size.
DO terminal type	WILL terminal type	Can we negotiate terminal type?
SB terminal type SE		Send me your terminal characteristics.
	SB terminal type IBM=3278-2 SE	My terminal is a 3278-2.
DO echo	WONT echo	

- 3 octets pour transmettre une option :
 - <IAC> <command code> <option number>
 - ex : IAC WILL Terminal_type soit 255 251 24

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 30

30

Paramètrer la connexion (Commandes Telnet)

Pour passer en mode commande durant une connexion, il suffit de taper le caractère d'échappement

```
sh-2.05a$ telnet
telnet> ?
Commands may be abbreviated, Commands are:

close      close current connection
logout     forcibly logout remote user and close the connection
display    display operating parameters
mode       try to enter line or character mode ('mode ?' for more)
open       connect to a site
quit       exit telnet
send       transmit special characters ('send ?' for more)
set        set operating parameters ('set ?' for more)
unset      unset operating parameters ('unset ?' for more)
status     print status information
toggle     toggle operating parameters ('toggle ?' for more)
slc        set treatment of special characters

z          suspend telnet
environ   change environment variables ('environ ?' for more)
telnet> status
No connection.
Escape character is '^]'.
telnet>
```

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

31

31

RLOGIN : principe

Remote LOGIN (service login dans inetd.conf)

- Application standard d'unix BSD (RFC 1282) (dec 1991)
- Connexion TCP sur le port 513 côté serveur
- Plus simple que telnet (que sous Unix)
- Idée : lors de la connexion, les paramètres du terminal local sont envoyés au site distant (pas de négociation)
- Intérêts de rlogin par rapport à telnet
 - permet à l'administrateur de définir un ensemble de machines "équivalentes" sur lesquelles les noms d'utilisateurs et les droits d'accès sont partagés
 - exemple : un utilisateur a un login X sur m1 et Y sur m2
 - permet des accès automatiques sans saisir de mot de passe
 - permet d'exporter sur la machine distante une partie de l'environnement local (type du terminal \$TERM, taille de la fenêtre) : un terminal distant a alors un comportement similaire à un terminal local (couleurs...)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

32

32

RLOGIN : authentification

- Authentification
 - si un mot de passe est nécessaire, il circule en clair
- Authentification automatique
 - pour ne pas avoir à saisir de mot de passe, il faut
 - soit que la machine cliente soit dans le fichier /etc/hosts.equiv de la machine distante
 - soit que le couple (machine cliente, user) soit dans le fichier \$HOME/.rhosts de la machine distante
 - le démon rlogind examine d'abord si le fichier /etc/hosts.equiv permet une authentification automatique, puis si tel n'est pas le cas, il regarde le fichier \$HOME/.rhosts

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

33

33

RLOGIN : authentification

- le fichier \$HOME/.rhosts permet d'éviter l'authentification de certains couples (machine cliente/utilisateur)

```
ogluck@192.168.69.1# cat .rhosts
192.168.69.2 ogluck
```
- le fichier /etc/hosts.equiv contient les machines "équivalentes" ou des entrées de type .rhosts

```
ogluck@192.168.69.2# cat /etc/hosts.equiv
192.168.69.1 # autorise tout le monde
192.168.69.1 ogluck # que ogluck
+ ogluck # ogluck depuis n'importe où
```

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

34

34

RSH : principe

Remote SHell

- Connexion TCP port 514 - le pendant de rlogin
- Exécution de commandes sur une machine distante de façon transparente

```
rsh host cmd
```

 - authentification automatique comme avec rlogin
 - tout se passe comme si l'exécution était locale
 - l'entrée standard et la sortie standard de cmd sont directement connectées à la socket cliente
 - avantage : peut être utilisé directement dans un programme (pas de saisie de mot de passe)
 - quand cmd se termine sur le site distant, le processus rsh client se termine
 - une séquence Ctrl-C termine le processus distant cmd

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

35

35

RSH : principe

- Exemple

```
ogluck@192.168.69.1# rsh 192.168.69.2 ls
interfaces
iperf-1.7.0
iperf-1.7.0-source.tar.gz
iperf.deb
```
- Fonctionnement du démon rshd quand une requête arrive
 - 1- lecture sur la socket jusque '\0' (octet nul) ; la chaîne lue est interprétée comme un numéro de port
 - 2- une deuxième connexion est établie vers le client vers ce numéro de port pour transmettre stderr (permet de distinguer stderr et stdout dans les >)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

36

36

RSH : principe

- 3- récupération de l'@ IP cliente pour déterminer un nom éventuel (requête DNS) pour l'authentification
- 4- lecture sur la socket initiale
 - du username sur la machine cliente (`user_l`)
 - du username sur la machine distante (`user_d`)
 - de la ligne de commande à exécuter
- 5- le démon authentifie l'exécution distante
 - il vérifie que `user_d` est bien dans `/etc/passwd`
 - si `user_l=user_d`, regarde dans `/etc/hosts.equiv`
 - sinon regarde dans `$HOME/.rhosts`
- 6- une fois `user_d` authentifié, le démon renvoie '\0' au client puis passe la ligne de commande au shell local

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 37

37

RSH : principe

Requête cliente (socket initiale)

rsch client → \0 ls \0 ogluck \0 ogluck \0 1048 → rshd

Ouverture d'une deuxième socket

Authentification { /etc/passwd, /etc/hosts.equiv, /home/ogluck/.rhosts }

Exécution de la commande

clavier → socket initiale → stdin → 1s

écran → deuxième socket → stdout, stderr → 1s

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 38

38

SSH : un shell distant sécurisé

Secure Shell

- Les communications sont cryptées
- Authentification à base de clés
- Un des seuls protocoles de connexion à distance qui passe les pare-feux de nos jours
- Permet de transporter des fenêtres graphiques via le tunnel SSH (multiplexage de plusieurs flux dans la connexion)
- Connexion TCP sur le port 22 côté serveur
- Pas encore de RFC (ietf-internet-draft)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 39

39

SSH : syntaxes

- Connexions à distance (style `rlogin`)


```
ssh -l user hostname
```

```
ssh user@hostname
```
- Exécution de commande à distance (style `rsh`)


```
ssh -l user hostname cmd
```

```
ssh user@hostname cmd
```
- Copie de fichiers à distance (style `rcp`)


```
scp file1 file2 user@hostname:
```

```
scp -r dir user@hostname:/tmp
```
- ssh et scp remplacent `rlogin`, `rsh`, `rcp`, ...

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 40

40

SSH : tunnels (port forwarding)

- `telnet pop.univ-lyon1.fr 110` ---> non sécurisé !
- On peut faire passer n'importe quoi dans un tunnel ssh :


```
ssh -L 3280:pop.univ-lyon1.fr:110 pop.univ-lyon1.fr
```

```
telnet localhost 3280
```
- Tout ce qu'on envoie sur `localhost/3280` arrive sur `pop/110` mais les données sont chiffrées (encapsulées dans le protocole ssh)

localhost:3280 → 1538 | 22 → Internet → 22 → pop:110

Données chiffrées par ssh

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 41

41

SSH : chiffrement

- Principe du chiffrement


```
Message en clair → f(msg, clé) → Message chiffré
```

Clé de chiffrement

Algorithme de chiffrement
- La qualité de la sécurité dépend
 - du secret de la clé
 - de la longueur de la clé (plus il y a de bits, plus il est difficile d'essayer toutes les clés)
 - de la difficulté d'inversion de l'algorithme de chiffrement

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 42

42

SSH : chiffrement

- Deux types d'algorithmes
 - symétrique** : même clé privée secrète partagée utilisée pour le chiffrement et le déchiffrement
 - l'émetteur et le récepteur doivent se mettre d'accord sur la clé à utiliser
 - asymétrique** : utilisation d'une clé publique pour le chiffrement et d'une clé privée pour le déchiffrement

$m_{chiffre} = f(m_{clair}, C_{publique})$ et $m_{clair} = g(m_{chiffre}, C_{privée})$

- pour qu'un émetteur envoie un message chiffré, il suffit qu'il connaisse la clé publique du destinataire
- pb : comment être sûr que la clé publique est bien celle du destinataire escompté ?
- certificat : association d'une clé publique et d'un nom de destinataire signée par un tiers de confiance

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 43

43

Chiffrement symétrique et asymétrique

source : M. Herrb

Symétrique : DES / AES

Assymétrique : RSA (DSA - El-Gamal)

Le client ssh Bob envoie sa clé publique à Alice qui l'utilise pour chiffrer la clé secrète

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 44

44

SSH : chiffrement

- Dans SSH :
 - algorithme asymétrique pour l'authentification (généralement RSA : basé sur l'arithmétique modulo)
 - algorithme symétrique pour les communications
 - utilisation de RSA pour échanger la clé de l'algorithme symétrique
 - chiffrement et déchiffrement moins coûteux

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 45

45

Chiffrement pratique

source : M. Herrb

Fonctions à clé publique très coûteuses → utilisation d'une clé de session

Chiffrement par Alice :

ssh de Bob vers Alice, Bob envoie sa clé publique à Alice, Alice fabrique une clé secrète (la clé de session) pour l'envoyer chiffrée à Bob. C'est cette clé secrète qui sera utilisée ensuite pour chiffrer/déchiffrer de manière symétrique avec AES.

Longueur des clés sûres (2003) :
 clé secrète : 128 bits
 clé publique/privée : 1024 bits

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 46

46

Déchiffrement pratique

source : M. Herrb

Déchiffrement par Bob :

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 47

47

SSH : authentification

- 4 méthodes sont essayées dans l'ordre par sshd
 - authentification automatique (souvent désactivée sur le serveur car considérée *insecure*)
 avec `/etc/hosts.equiv` ou `/etc/ssh/hosts.equiv`
 avec `~/.rhosts` ou `~/.shosts`
 - authentification automatique forte "améliorée"
 idem (avec fichiers `rhosts` ou `hosts.equiv`) mais combinée avec une authentification des *hosts* par RSA : le serveur vérifie la clé en provenance de la machine cliente
 Mot de passe demandé lors de la première connexion puis ajout du couple (@IP, clé publique) du serveur dans le fichier du client
 Si la clé publique change pour cette @IP alors on redemande une authentification forte par mot de passe
`/etc/ssh/ssh_known_hosts` ou `~/.ssh/ssh_known_hosts`

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 48

48

SSH : authentification

- 4 méthodes sont essayées dans l'ordre par `sshd`
 - authentification automatique forte par RSA (asymétrique)
 - le client génère un couple (clé_pub, clé_pri) avec `ssh-keygen` et copie la clé publique dans le fichier `~/.ssh/authorized_keys` sur le serveur ; liste les clés publiques autorisées (équivalent du `~/.rhosts` !)
 - le client envoie au serveur sa clé publique ; si elle est dans le fichier, le serveur génère un nb aléatoire de 256 bits qu'il chiffre avec la clé envoyée par le client ; le client déchiffre avec sa clé privée puis la renvoie avec hachage MD5 ; le serveur calcule le hachage MD5 et vérifie
 - authentification par saisie du mot de passe mais ce dernier est chiffré par RSA avant d'être envoyé au serveur

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 49

49

X : multi-fenêtrage réparti

- Système de multi-fenêtrage sous Unix
 - appelé X ou X Window System ou X11
 - ensemble de programmes réalisant l'interface Homme/Machine basé sur l'utilisation des périphériques (clavier, souris, écran, ...)
- X est constitué de plusieurs entités
 - un serveur X : gère le matériel (clavier, écran, ...) et leur utilisation par les applications graphiques ; accessible sur le port TCP $6000+n$ où n est le numéro de DISPLAY
 - des clients X : applications graphiques qui nécessitent un serveur X (`xemacs`, `xterm`, `xcalc`, `xv`, ...)
 - le protocole X : fait communiquer les clients et le serveur

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 50

50

X : multi-fenêtrage réparti

- Système réparti : permet de travailler sur plusieurs machines simultanément
 - les clients X peuvent s'exécuter sur des machines distantes (3 connexions TCP dans l'exemple)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 51

51

X : multi-fenêtrage réparti

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 52

52

X : multi-fenêtrage réparti

- Chaque client X peut définir ses caractéristiques
 - spécifications standards
 - fontes, géométrie de la fenêtre, background, foreground, borderwidth, couleurs...
 - spécifications particulières à l'application
 - affichage ou non d'un ascenseur...
- Gestion de fenêtres : *Window Manager*
 - un client X particulier qui gère
 - déplacement/redimensionnement de fenêtre
 - créer/détruire/iconifier des fenêtres
 - lancer ou terminer des applications X

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 53

53

X : multi-fenêtrage réparti

- Le protocole X permet au serveur X de contrôler l'autorisation des accès
 - Quels clients X peuvent se connecter au serveur X ?
 - La commande `xhost`

```

ogluck@lima:~$ xhost
access control enabled,only authorized clients can connect
ogluck@lima:~$ echo $DISPLAY
140.77.13.102:0.0
ogluck@lima:~$ xhost + ble
ble being added to access control list
ogluck@lima:~$ rlogin ble
ogluck@ble:~$ export DISPLAY=140.77.13.102:0.0
ogluck@ble:~$ xterm &
ogluck@ble:~$ exit
Connection to ble closed.
ogluck@lima:~$ xhost - ble
ble being removed from access control list
    
```

Qui est le serveur X ?

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 54

54

SSH : X11 et TCP forwarding

- X11 Forwarding
 - permet d'avoir une encapsulation chiffrée du protocole X11 dans la connexion `ssh` avec une gestion automatique de la variable `$DISPLAY`
 - si la variable `$DISPLAY` du client `ssh` est positionnée, `ssh -X` permet au serveur d'exporter les fenêtres graphiques lancées à partir de la connexion `ssh` vers le `$DISPLAY` du client (un "proxy X server" est créé sur la machine serveur pour transférer les connexions X vers le client via la session `ssh`)
- Possibilité de rediriger n'importe quel port TCP (dépend de la configuration de `ssh`)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 55

55

SSH : X11 forwarding

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 56

56

Applications de transfert de fichiers

Protocoles de transfert de fichiers

FTP : File Transfer Protocol

TFTP : Trivial File Transfer Protocol

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 57

57

Protocoles de transfert de fichiers

- Copie intégrale d'un fichier d'un système de fichiers vers un autre en environnement hétérogène
- L'hétérogénéité concernant les fichiers est dépendante d'un système à l'autre
 - de la façon de représenter les noms de fichier (longueur, caractère espace,...)
 - des droits d'accès au fichier (lecture, écriture, exécution, propriétaire, ...)
 - de la représentation des données contenues dans le fichier (saut de ligne...)
 - --> mode `ascii` : transfert au format NVT avec conversion au format local (`TYPE A`)
 - --> mode `binary` : transfert sans conversion (`TYPE I`)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 58

58

Protocoles de transfert de fichiers

- Plusieurs protocoles
 - copie de fichiers à distance : `rsh`, `scp`
 - protocole de transfert de fichiers avec accès aux systèmes de fichiers local et distant : `ftp`, `tftp`, `sftp`
- Type client/serveur
 - le client (initiateur de la connexion) interagit avec l'utilisateur, le système de fichiers local et les protocoles réseau
 - le serveur (héberge les fichiers distants) interagit avec les protocoles réseau et le système de fichiers distant
- Ne pas confondre avec les protocoles d'accès aux fichiers distants : NFS (RPC), SMB (Microsoft)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 59

59

Protocoles de tranfert de fichiers

- Transfert de fichiers vers/depuis un hôte distant

~~Machine cliente~~ ~~Machine distante~~

~~Nous ne pouvons pas afficher l'image~~

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 60

60

Interfaces utilisateur

The screenshot shows the WinSCP application interface. At the top, there are two panes for 'Local System' and 'Remote System', both displaying file listings with columns for Name, Date, Size, and Rights. Below these is a 'Propriétés de connexion' (Connection Properties) window with fields for Host Name/Address, Host Type, User ID, Password, and Account. At the bottom, a terminal window shows the execution of 'ls' and 'ls -la' commands on the remote system.

61

FTP : File Transfer Protocol - RFC 959

- Standard TCP/IP pour le transfert de fichiers
- Connexion TCP sur le port 21 côté serveur
- Contrôle d'accès au serveur distant (login,mdp)
 - le mot de passe circule en clair
- Particularité de FTP par rapport à TELNET... : utilisation de 2 connexions TCP

```

ogluck@lima:~$ cat /etc/services | grep ftp
ftp-data      20/tcp
ftp           21/tcp
tftp          69/udp
sftp          115/tcp # FTP over SSH
ftp-data      989/tcp # FTP over SSL (data)
ftps          990/tcp # FTP over SSL
    
```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux

62

FTP : Connexions contrôle et données

- Les clients FTP contactent le serveur FTP sur le port TCP/21
- Ouverture de 2 connexions TCP parallèles :
 - Contrôle : échange des commandes et des réponses entre le client et le serveur - "contrôle hors-bande"
 - Données : transfert des fichiers de données vers/depuis l'hôte distant (sur le port TCP/20 côté serveur)
- Le serveur FTP maintient un "état" : répertoires courants local et distant, username

The diagram illustrates the connection between a Client FTP and a Server FTP. Both are connected to the Internet via TCP/IP. The control connection is established on port 21, and the data connection is established on port 20. The client sends commands like 'put' and 'get' over the control connection, and data is transferred over the data connection.

63

FTP : Connexions contrôle et données

- Scénario d'une connexion
 - le client FTP se connecte sur le port 21 du serveur
 - le port 21 sert à envoyer des commandes au serveur FTP (put, get, cd, ...)
 - si une commande nécessite que des données soient reçues ou transmises (ls, get, put, ...), le client envoie une commande PORT au serveur indiquant un port (p1*256+p2) sur lequel le serveur va créer une connexion ftp-data depuis son port 20
 - la connexion ftp-data est close dès que le transfert est terminé

The sequence diagram shows the interaction between a Client and a Server. The client connects to the server on port 21 for 'Connexion, authentification (USER, PASS)'. The client then sends a 'PORT h1,h2,h3,h4,p1,p2 (@IP, n° port)' command. The server responds with 'LIST'. The client then sends 'data' to the server on port 20. The client's local port is shown as 33217.

64

FTP : Connexions contrôle et données

- Connexion contrôle (ftp) :
 - échange des requêtes/réponses (dialogue client/serveur)
 - permanente, full-duplex, besoin de fiabilité (et faible délai !)
 - initiée par le client
- Connexion données (ftp-data) :
 - envoi de fichier ou liste de fichiers/répertoires (données)
 - temporaire, full-duplex, besoin de débit (et fiabilité !)
 - initiée par défaut par le serveur
 - ouverture active (connect()) du serveur vers le client (depuis le port 20 vers le port ??)
 - la connexion est fermée dès que le caractère EOF est lu

Olivier Glück M2 SRS - Admin Systèmes et Réseaux

65

FTP : Connexions contrôle et données

- Active transfer & Passive transfer
 - Active transfer : la connexion ftp-data est initiée par le serveur
 - > problème de firewall ou de NAT : impossibilité de créer la connexion à partir du serveur même s'il connaît le numéro de port du client
 - Passive transfer : ftp-data initiée par le client
 - intégré dans les navigateurs, paramétrable sur certains clients
 - fonctionnement : le client envoie la commande PASV au lieu de PORT sur le port 21 (RFC 1579 : Firewall-Friendly FTP) ce qui revient à demander au serveur de faire un listen() sur le port 20

Olivier Glück M2 SRS - Admin Systèmes et Réseaux

66

Commandes du client FTP

- Ne pas confondre avec les commandes du protocole FTP !

```

xterm
ftp> help
Commands may be abbreviated.  Commands are:

!          debug      nmdir      qc          send
$          dir         nget      sendport   site
account   disconnect  nkdir     put        status
append    exit         nis       pwd        struct
ascii    form         mode      quit       system
bell      get          modtime   quote     system
binary   glob        mput     recv      sunique
bye      hash       newer    reget     tenex
case     help       nmap     rstatus   tick
cd       idle      nlist    rhelp     trace
cdup    image     ntrans  rename   type
chmod   lcd       open     reset    user
close   ls        prompt  restart  unask
cr      makedirs passive  rmdir    verbose
delete  delete    proxy   runique  ?

ftp> help passive
passive      enter passive transfer mode
ftp> help open
open         connect to remote ftp
ftp> help get
get         receive file
ftp>
    
```

Varie d'un client à l'autre !



67

Requêtes du protocole FTP

```

xterm
ogluck@lima:~$ telnet localhost 21
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 lima.cri2000.ens-lyon.fr FTP server (Version 6.4/OpenBSD/Linux-ftp-0.17) ready.
HELP
214- The following commands are recognized (* =>'s unimplemented).
USER PORT STOR MSAM* RNTD NLST MKD CUP
PHSS PHSV APPE MRSQ* ABOR SITE XMKD XCUP
ACCT* TYPE MLFL* MRCF* DELE SYST RMD STOU
SHNT* STRU MAIL* ALLO CWD STAT XRMN SIZE
REIN* MODE MSND* REST XCMD HELP PWD MDTH
QUIT RETR MSDM* RNFR LIST NOOP XPWD
214 Direct comments to ftp-bugs@lima.cri2000.ens-lyon.fr.
HELP PHSV
214 Syntax: PHSV (set server in passive mode)
USER ogluck
331 Password required for ogluck.
PHSS secret!
230- Linux lima 2.4.22-1-686 #6 Sat Oct 4 14:09:08 EST 2003 1686 GNU/Linux
230 User ogluck logged in.
PHSV
227 Entering Passive Mode (127,0,0,1,133,57)
LIST
    
```

ABOR : interrompt le transfert en cours (à la suite d'un ctrl-c lors d'un transfert)

Pourquoi rien ne s'affiche ???

68

Requêtes du protocole FTP

RETR <filename>

Déclanche la transmission par le serveur du fichier <filename> sur le canal de données.

STOR <filename>

Déclanche la réception d'un fichier qui sera enregistré sur le disque sous le nom <filename>. Si un fichier avec le même nom existe déjà il est remplacé par un nouveau avec les données transmises.

APPE <filename>

Déclanche la réception d'un fichier qui sera enregistré sur le disque sous le nom <filename>. Si un fichier avec le même nom existe déjà, les nouvelles données lui sont concaténées.

REST <offset>

Redémarrage en cas d'échec d'un transfert précédent. L'offset précise le numéro du dernier octet reçu.

ABOR : abandon d'un transfert en cours.

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

69

Requêtes du protocole FTP

PWD : impression du répertoire courant.

LIST : catalogue du répertoire courant (canal donnée).

NLST : catalogue succinct (canal donnée).

CWD <repname> : changement de répertoire courant pour <repname>.

MKD <repname> : création du nouveau répertoire <repname>.

RMD <repname> : suppression du répertoire <repname>.

DELE <filename> : suppression du fichier <filename>.

RNFR <filename1> : définit le nom actuel d'un fichier à renommer.

RNTO <filename2> : définit le nouveau nom d'un fichier à renommer.

STAT : status courant de la session FTP.

STAT <repname> : équivalent à LIST mais réponse sur le canal de contrôle.

HELP : affiche l'aide sur les opérations du site.

NOOP : no operation.

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

70

Réponses du protocole FTP

- Les réponses sont de la forme

status_code description

Le code est un nombre sur trois chiffres signifiant :

Status	Signification	Status	Signification
1yz	Réponse positive préliminaire (une autre réponse suivra)	x0z	Erreur de syntaxe
2yz	Réponse positive finale (une autre requête est possible)	x1z	Réponse Informativ (HELP...)
3yz	Réponse positive intermédiaire (une autre requête doit suivre)	x2z	Relatif à une connexion
4yz	Réponse négative temporaire (la même requête peut réussir plus tard)	x3z	Relatif à l'authentification
5yz	Réponse négative définitive (la requête n'est pas acceptée)	x5z	Relatif au système de fichier

71

Exemples de réponses

- 125 Data connection already open
- 150 Opening BINARY mode data connection
- 200 Command successful
- 214 Help message
- 220 lima.cri2000.ens-lyon.fr FTP server (Version 6.4/OpenBSD/Linux-ftp-0.17) ready
- 226 Transfer complete
- 230 User ogluck logged in
- 331 Passwd required for ogluck
- 425 Can't open data connection
- 452 Error writing file
- 500 Command not understood
- 550 No files found

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

72

71

72

Exemple de dialogue FTP

```

ogluck@lima:~$ ftp -d -v localhost
Connected to localhost.
220 lima.cri2000.ens-lyon.fr FTP server (Version 6.4/OpenBSD/Linux-ftp@0.17) ready.
Name (localhost:ogluck): ogluck
--> USER ogluck
331 Password required for ogluck.
Password:
--> PASS XXXX
230 Linux lima 2.4.22-1-686 #6 Sat Oct 4 14:09:08 EST 2003 i686 GNU/Linux
230 User ogluck logged in.
--> SYST
215 UNIX Type: L8 (Linux)
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> lcd /tmp
Local directory now /tmp
ftp> cd FTP
--> CWD FTP
250 CWD command successful.
ftp> ls
--> PORT 127,0,0,1,133,83
200 PORT command successful.
--> LIST
150 Opening ASCII mode data connection for '/bin/ls'.
Total 4
-rw-r--r--  1 ogluck      2692 Mar 12 17:12 ftp.log
226 Transfer complete.

```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 73

73

Exemple de dialogue FTP

```

ftp> get ftp.log
local: ftp.log remote: ftp.log
--> TYPE I
200 Type set to I.
--> PORT 127,0,0,1,133,84
200 PORT command successful.
--> RETR ftp.log
150 Opening BINARY mode data connection for 'ftp.log' (2692 bytes).
226 Transfer complete.
2692 bytes received in 0.00 secs (45326.0 kB/s)
ftp> put ftp.log ftp2.log
local: ftp.log remote: ftp2.log
--> PORT 127,0,0,1,133,85
200 PORT command successful.
--> STOR ftp2.log
150 Opening BINARY mode data connection for 'ftp2.log'.
226 Transfer complete.
2692 bytes sent in 0.00 secs (90651.9 kB/s)
ftp> mkdir TOTO
--> MKD TOTO
257 "TOTO" directory created.
ftp> ls
--> TYPE A
200 Type set to A.
--> PORT 127,0,0,1,133,86
200 PORT command successful.

```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 74

74

Exemple de dialogue FTP

```

--> LIST
150 Opening ASCII mode data connection for '/bin/ls'.
Total 12
drwxr-x---  2 ogluck      4096 Mar 12 17:17 TOTO
-rw-r--r--  1 ogluck      2692 Mar 12 17:12 ftp.log
-rw-r--r--  1 ogluck      2692 Mar 12 17:16 ftp2.log
226 Transfer complete.
ftp> passive
Passive mode on.
ftp> get ftp2.log
local: ftp2.log remote: ftp2.log
--> TYPE I
200 Type set to I.
--> PASV
227 Entering Passive Mode (127,0,0,1,133,87)
--> RETR ftp2.log
150 Opening BINARY mode data connection for 'ftp2.log' (2692 bytes).
226 Transfer complete.
2692 bytes received in 0.00 secs (53651.1 kB/s)
ftp> bye
--> QUIT
221 Goodbye.
ogluck@lima:~$

```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 75

75

FTP : autorisation d'accès

- Contrôle fin possible au niveau de chaque utilisateur : le fichier /etc/ftpusers permet d'empêcher les connexions FTP de certains utilisateurs

```

ogluck@lima:~$ cat /etc/ftpusers
# /etc/ftpusers: list of users disallowed
ftp access. See ftpusers(5).
root
ftp
anonymous

```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 76

76

TFTP : Trivial File Transfer Protocol

- Transfert de fichiers au-dessus d'UDP, port 69

```

ogluck@lima:~$ grep tftp /etc/services
tftp          69/udp
ogluck@lima:~$ grep tftp /etc/inetd.conf
tftp         dgram  udp    wait  nobody
           /usr/sbin/tcpd  /usr/sbin/in.tftpd /tftpdboot

```
- Pourquoi TFTP ?
 - TFTP, c'est en gros FTP sans pouvoir lister les répertoires distants et ne nécessitant pas de mot de passe pour récupérer ou déposer des fichiers
 - protocole léger donc facilement implantable par des systèmes sans disque (en ROM) qui utilisent TFTP au boot pour récupérer un fichier de configuration... (terminaux X, imprimantes réseau, routeurs Cisco...)
 - UDP car ces systèmes n'implémentent pas forcément TCP

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 77

77

TFTP : Trivial File Transfer Protocol

- Protocole léger - RFC 1350
 - pas de contrôle d'accès
 - 5 types de messages seulement
 - fiabilité assurée par acquittement positif avec timer de retransmission sur l'émetteur et le récepteur
 - les messages DATA font 512 octets max ; ils sont numérotés et sont aussitôt acquittés par un ACK
- Comme il n'y a pas d'authentification, les accès sur le serveur sont limités aux répertoires passés en arguments du démon tftpd (/tftpdboot par défaut)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 78

78

TFTP : types de messages

- Les 5 types de messages

Opcode	Opération	Description
1	RRQ	Read request
2	WRQ	Write request
3	DATA	Data
4	ACK	Acknowledgment
5	ERROR	Error (sert aussi d'ACK)

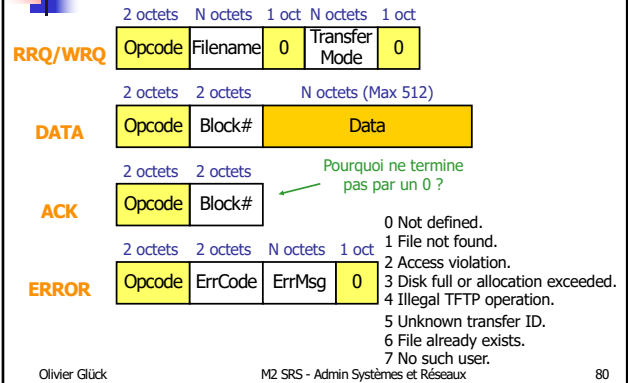
Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

79

79

TFTP : format des messages



Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

80

80

TFTP : commandes utilisateurs

```

lima
Olivier Glück:~/FTP$ tftp
tftp> ?
Commands may be abbreviated.  Commands are:
connect      connect to remote tftp
mode         set file transfer mode
put          send file
get          receive file
quit         exit tftp
verbose      toggle verbose mode
trace        toggle packet tracing
status       show current status
binary       set mode to octet
ascii        set mode to netascii
recv        set per-packet retransmission timeout
timeout      set total retransmission timeout
?            print help information

tftp> verbose
Verbose mode on.
tftp> trace
Packet tracing on.
tftp> get toto
getting from localhost:toto to toto [netascii]
sent RRQ <file=toto, mode=netascii>
received DATA <block=1, 0 bytes>
tftp> put toto
putting toto to localhost:toto [netascii]
sent WRQ <file=toto, mode=netascii>
received ERROR <code=2, msg=Access violation>
Error code 2: Access violation
tftp> put toto2
putting toto to localhost:toto2 [netascii]
sent WRQ <file=toto2, mode=netascii>
received ERROR <code=1, msg=File not found>
Error code 1: File not found
tftp>

lima /tftpboot
lima-/tftpboot# ls -l
total 0
-rw-r--r-- 1 root root 0 Mar 14 18:30 toto
lima-/tftpboot#
    
```

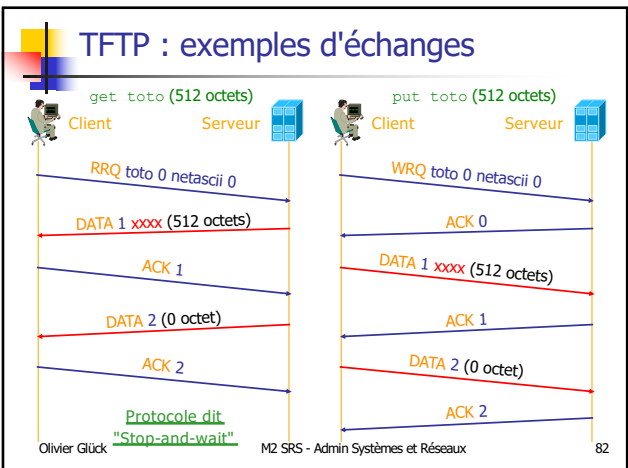
Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

81

81

TFTP : exemples d'échanges



Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

82

82

Accès aux fichiers distants

Les protocoles NFS et SMB



Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

83

83

Accès aux fichiers distants

- Différences avec le transfert de fichiers
 - l'accès aux fichiers distants est complètement transparent pour l'utilisateur
 - tout se passe comme si le système de fichiers distant était local
 - l'utilisateur peut éditer le fichier, le modifier, ... ; les modifications seront répercutées sur le système de fichiers distant
- Les deux principaux protocoles
 - NFS : *Network File System* (Unix/Sun-RPC)
 - SMB : *Server Message Block* (issu du monde Microsoft)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

84

84

NFS : Network File System

- Présenté par Sun en 1985 pour permettre à ses stations sans disque d'accéder à un système de gestion de fichiers distant (RFC 1094)
- Utilise les appels de procédures distantes Sun-RPC (qui sont issues des travaux sur NFS)
 - a priori, les client et serveur NFS devraient être des processus utilisateur s'exécutant au-dessus de RPC/XDR/UDP/IP
 - en fait, le client et serveur NFS s'exécutent dans le noyau
 - le client pour rendre transparent l'accès à un fichier via NFS (pas de différence d'utilisation d'un fichier local et d'un fichier distant)
 - le serveur pour des raisons d'efficacité

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 85

85

NFS : principe de fonctionnement

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 86

86

Les éléments d'accès aux fichiers

- Manipule des chemins, descripteurs, déplacements
- Manipule des Files, Dentries, Inodes, déplacements
- Masque les différences à l'application (API uniforme, ...)
- Manipule des blocks
- Matériel-dépendant

Le choix entre ext4, NTFS, VFAT, NFS, ... se fait lors de l'ouverture du fichier

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 87

87

NFS : en pratique...

```

root@192.168.69.1# tail -1 /etc/fstab
192.168.69.2:/home /nfshome nfs defaults,noauto 0 0
root@192.168.69.1# mkdir /nfshome
root@192.168.69.1# mount /nfshome
root@192.168.69.1# ls -l /nfshome
drwxr-xr-x 2 1003 5000 1024 fév 16 13:32 olivier
root@192.168.69.1# df
Filesystem            1k-blocks    Used Available Use% Mounted on
/dev/hda2              3898108    2766592  930304   75% /
192.168.69.2:/home    16128636   1493328  13815996  10% /nfshome
  
```

Client NFS

```

root@192.168.69.2# cat /etc/exports
# /etc/exports: the access control list for filesystems which
# may be exported to NFS clients. See exports(5).
/home 192.168.69.0/255.255.255.0(rw,root_squash,async)
  
```

Server NFS

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 88

88

NFS : en pratique...

```

root@192.168.69.1# rpcinfo -u 192.168.69.2 showmount
program 100005 version 3 ready and waiting
root@192.168.69.1# rpcinfo -u 192.168.69.2 nfs
program 100003 version 3 ready and waiting
root@192.168.69.1# showmount -a 192.168.69.2
All mount points on 192.168.69.2:
192.168.69.1:/home
root@192.168.69.1# showmount -d 192.168.69.2
Exported directories on 192.168.69.2:
/home
root@192.168.69.1# showmount -e 192.168.69.2
Export list for 192.168.69.2:
/home 192.168.69.0/255.255.255.0
  
```

Client NFS

```

root@192.168.69.2# showmount
Hosts on 192.168.69.2:
192.168.69.1
  
```

Server NFS

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 89

89

NFS et la sécurité

- Principe d'authentification :
 - (uid, gid)_{local} est mappé en (uid, gid)_{distant} --> équivalence entre les droits locaux et distants
 - problème pour root :
 - quels droits possède le root d'une machine cliente sur les fichiers exportés par un serveur NFS ?
 - par défaut, root est mappé en nobody pour des raisons de sécurité (sinon il faut mettre no_root_squash dans /etc/exports)
- Règle de non transitivité de NFS :
 - si A exporte /home à B ; si B monte A : /home dans /home2 et exporte /home2 à C alors C n'aura pas accès au /home de A
 - sinon il n'y aurait aucun contrôle possible des règles d'exportation donc pas de sécurité... (B pourrait exporter à C alors que A ne veut pas autoriser C)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 90

90

NFS : les procédures distantes (v2)

- GETATTR : retourne les attributs d'un fichier (type, taille, permissions, propriétaire, date de dernière modification, ...)
- SETATTR : modifie les attributs d'un fichier
- STATFS : retourne l'état d'un SGF (espace libre, ...)
- LOOKUP : recherche le fichier dont le nom et en argument, retourne un *file handle* et les attributs (appelée à l'ouverture du fichier)
- READ : lecture dans le fichier d'au plus N octets à partir d'un certain offset (N doit être inférieur à 8192)
- WRITE : écriture dans le fichier de N octets à partir d'un certain offset, les données se trouvant dans le tampon @
- CREATE/REMOVE/RENAME : créer, supprimer, renommer
- LINK/SIMLINK/READLINK : création/lecture d'un lien
- MKDIR/RMDIR/READDIR : manipulation des répertoires

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

91

91

MOUNT : les procédures distantes

- Procédures distantes utilisées par `mount` et `showmount`
 - MNT : montage d'une partition distante, retourne un *file handle* correspondant au répertoire monté (racine)
 - DUMP : retourne la liste de tous les *file system* montés
 - UMNT : supprime un point de montage
 - UMNSTALL : supprime tous les points de montage pour ce client
 - EXPORT : retourne les informations sur les *file system* qui sont exportés

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

92

92

NFS : procédures non-idempotentes

- Toutes les opérations précédentes ne sont pas idempotentes : deux exécutions identiques successives ne donnent pas le même résultat
- Ne sont pas idempotentes `CREATE`, `REMOVE`, `RENAME`, `LINK`, `SIMLINK`, `MKDIR`, `RMDIR`
- Solution :
 - le serveur utilise un cache des requêtes/réponses récentes pour les procédures non-idempotentes
 - quand une nouvelle requête arrive, le serveur regarde dans son cache si la réponse est déjà présente, auquel cas il renvoie cette réponse sans re-exécuter la proc.

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

93

93

NFS : un serveur sans état

- Le serveur NFS est sans état : entre deux requêtes, il ne conserve aucune information sur
 - les accès précédents à un fichier donné,
 - les fichiers ouverts, ...
 - le `LOOKUP` correspond au `open()` mais il n'y a pas de procédure correspondant au `close()`
 - après un `LOOKUP`, le serveur ne sait pas si le client va effectivement "utiliser" le fichier ou non
- Pourquoi sans état ?
 - en cas de crash du serveur NFS
 - permet de simplifier son redémarrage
 - transparent pour le client : il n'a pas besoin de réitérer certaines requêtes en cas de crash

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

94

94

NFS au dessus de TCP

- En principe, pour plus de réactivité, NFS utilise les RPC sur UDP (dans le cadre d'un LAN)
- Mais pour étendre NFS aux WAN, les dernières versions permettent d'exécuter NFS sur RPC/TCP
- Caractéristiques de NFS sur TCP
 - le serveur fait une ouverture passive sur le port TCP/2049
 - quand un client monte une partition NFS, cela se traduit par une ouverture active --> une connexion TCP par point de montage (soit une par *file system*)
 - toutes les applications qui utilisent ce système de fichiers partagent la même connexion TCP
 - plus résistant aux pannes du serveur NFS : le client essaie régulièrement de rétablir la connexion et conservent les requêtes RPC en attente

Olivier Glück

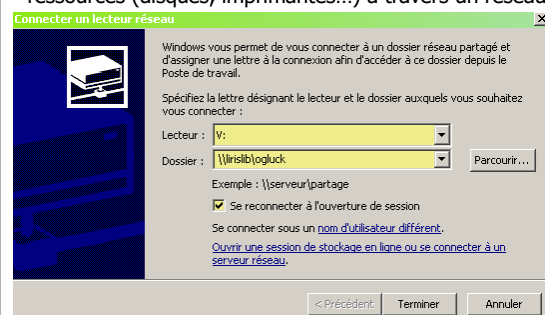
M2 SRS - Admin Systèmes et Réseaux

95

95

SMB : Server Message Block

- Protocole de Microsoft et Intel permettant le partage de ressources (disques, imprimantes...) à travers un réseau (1987)



96

96

SMB : Server Message Block

- SMB est prévu pour être utilisé sur l'interface NetBIOS
 - utilise les noms NetBIOS (15 caractères + 1 pour le type)
 - utilise le mécanisme datagram de NetBIOS par *broadcast* comme service de nommage (NOM-->@MAC, pas d'adresse de niveau 3)

Application		
SMB		
Netbios		
TCP/IP	NetBEUI	IPX/SPX
802.x	PPP	...

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 97

97

SMB : Server Message Block

- Chaque machine (client ou serveur) possède un nom sur 15 caractères
- SMB ajoute un 16ième caractère pour distinguer
 - les serveurs de fichiers et d'imprimantes, les clients, ...
- Notion de domaine
 - un ensemble d'utilisateurs (avec nom et mot de passe) et de serveurs (avec des droits d'accès)
 - un *primary domain server* contient la base de données des utilisateurs et de leurs mots de passe
- Un serveur partage une ou plusieurs ressources
 - fichiers, imprimantes, ...
 - à chaque triplet (domaine, serveur, ressource) correspond un nom unique `\\serveur\ressource_name`

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 98

98

SMB : Server Message Block

- Deux niveaux de protection des accès :
 - au niveau de chaque utilisateur : basé sur le nom des utilisateurs (user, passwd), permet de gérer l'accès aux ressources voire aux éléments d'une ressource
 - au niveau de chaque ressource : un mot de passe commun à tous les utilisateurs est associé à une ressource pour y autoriser l'accès
- CIFS : *Common Internet File System*
 - dernière version de SMB proposant un meilleur passage à l'échelle (extensibilité)
 - divulgaration du protocole SMB par Microsoft à l'IETF en 1996 sous ce nom --> a permis l'apparition de Samba

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 99

99

SMB : Server Message Block

- Samba : implémentation de SMB sous Unix qui permet un partage de ressources entre les mondes Unix et Windows ; Samba permet de
 - partager un disque Unix pour des machines Windows
 - accéder à un disque Windows depuis une machine Unix
 - partager une imprimante Unix pour des machines Windows
 - utiliser une imprimante Windows à partir d'un hôte Unix
- Le serveur Samba sur la machine Unix émule un domaine SMB

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 100

100

SMB : Server Message Block

- Commandes Unix liées au serveur Samba
 - smbpasswd : permet de changer le mot de passe d'un utilisateur SMB
 - smbclient : permet d'interroger un serveur samba depuis Unix

`smbclient -L host # liste les ressources offertes par le serveur`
`smbclient //host/ressource # permet l'accès à la ressource`

```

ogluck@lisa:~$ smbclient //lisa/ogluck
Password:
smb: \> help
?          allname      archive      blocksize    cancel
cd         chead        chown        del           dir
du         exit         get          help         history
lscd      link         lowercase   ls           mask
nd        nget        mkdir       none         nput
newer     open        print       printmode    prompt
put       pd          queue       quit        reget
rd        recurse    reget       rename       reput
rm        rmdir      setmode     setmode     setmode
runmode  rmdir      setmode     setmode     setmode
translate

smb: \> cd FTP
smb: \FTP> dir
.                D          0   Sun Mar 14 18:30:41 2004
..               D          0   Fri Mar 12 17:09:14 2004
ftp.log         -          0   Sun Mar 14 18:27:49 2004
ftp2.log        -          2832 Fri Mar 12 17:16:10 2004
TOTO           D          0   Fri Mar 12 17:17:00 2004
toto           D          0   Sun Mar 14 18:55:42 2004

smb: \FTP>
48380 blocks of size 262144, 39143 blocks available
  
```

Olivier Glück 101

101

Gestion d'utilisateurs distants

Lyon 1

102

NIS : un annuaire répliqué

- NIS : *Network Information System*
 - introduit par SUN en 1985 (*Yellow Pages* (yp) à l'origine)
 - n'est pas un standard de l'Internet mais est largement utilisé
 - une base de données distribuée qui permet le partage d'informations système (*/etc/passwd*, */etc/hosts*, ...)
- Objectif : réduire le temps d'administration d'un parc de machines
 - simplifier la gestion des comptes, des mots de passe et les travaux d'administration dans le monde Unix
 - typiquement, il suffit de créer un nouvel utilisateur sur le serveur NIS pour que chaque machine client NIS ait accès aux informations de *login* de cet utilisateur

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 103

103

NIS : un exemple courant

```

ogluck@192.168.90.2# grep ogluck /etc/passwd
ogluck:x:1001:1001:,,,:/home/ogluck:/bin/bash
  
```

```

root@192.168.69.2# showmount -a 192.168.90.1
All mount points on 192.168.90.1:
192.168.69.2:/home
root@192.168.69.2# su - ogluck
ogluck@192.168.69.2# ls
toto titi ...
  
```

```

ogluck@192.168.90.1# cd /home/ogluck
ogluck@192.168.90.1# ls
toto titi ...
  
```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 104

104

NIS : architecture

- Architecture : découpage en domaines
 - modèle Client/Serveur au dessus des SUN-RPC
 - un **domaine NIS** contient
 - un serveur NIS maître qui maintient les "maps" (informations contenues dans la base)
 - zéro, un ou plusieurs serveurs NIS esclaves :
 - permet de décharger le serveur principal et d'être plus résistant aux pannes
 - le maître réplique ses informations vers les serveurs secondaires
 - des clients NIS qui peuvent interroger les serveurs maître ou secondaires

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 105

105

NIS : architecture

- Seul le maître peut modifier une map
- Les slave server diffusent les maps sans pouvoir les modifier (diminue les problèmes de cohérence)

Mise à jour via RPC

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 106

106

NIS : en pratique...

- Les maps sont stockées sur le serveur dans */var/yp/nom-de-domaine*
- Quand le fichier source d'une map est modifié (sur le serveur), il faut régénérer la map associée et éventuellement propager les modifications aux serveurs NIS esclaves

Chaque map stocke des couples clé/valeur

Diffusion aux esclaves

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 107

107

NIS : en pratique...

- La commande *ypcat* permet de voir le contenu d'une map depuis n'importe quel client
- Un client NIS
 - doit réaliser un *binding* pour interroger le serveur NIS
 - il faut donner le nom de domaine et préciser la méthode de localisation du/des serveurs : *broadcast* (le premier qui répond !) ou désignation explicite d'un serveur
 - nom de domaine positionné par la commande *domainname* ou dans le fichier */etc/defaultdomain* ou en renseignant la variable *NISDOMAIN* dans */etc/...*
 - doit faire tourner le démon *ypbind* qui recherche régulièrement le serveur NIS approprié
 - ypbind -broadcast*
 - ypset nom-serveur-NIS ; ypbind*
 - ypwhich* permet d'afficher le nom du serveur NIS

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 108

108

NIS : en pratique...

■ Un client NIS

- est aussi configurable dans `/etc/yp.conf`

```
root@192.168.69.2# cat /etc/yp.conf
#plusieurs entrées de ce type sont possibles
domain grid5000 server 192.168.90.2
#domain grid5000 broadcast
```
- il faut indiquer à la fin de chaque fichier d'aller consulter les maps associées si nécessaire et mettre `compat` dans `/etc/nsswitch.conf` (man `nsswitch.conf`)

```
root@192.168.69.2# tail -1 /etc/passwd
+:::
root@192.168.69.2# tail -1 /etc/group
+:::
```

■ Un serveur NIS esclave doit faire tourner

- `ypserv` pour répondre aux requêtes de ses clients NIS
- `ypbind` s'il est lui-même un client NIS (pas obligatoire)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 109

109

NIS : en pratique...

■ Un serveur NIS maître doit faire tourner

- `ypserv` pour répondre aux requêtes de ses clients NIS
- `ypbind` s'il est lui-même un client NIS (pas obligatoire)
- `ypxfrd` pour répondre aux demandes de mise à jour des maps de la part des serveurs esclaves
- `rpc.yppasswd` pour assurer les demandes de changement de mots de passe (`yppasswd`)

■ Les netgroups (`/etc/netgroup`)

- le système NIS permet de définir des groupes de machines ou d'utilisateurs et ainsi d'autoriser ou non l'accès à une ressource
- le fichier de netgroup constitue une map
- un groupe est défini par une liste de triplets (machine, user, nis-domain)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 110

110

NIS : en pratique...

■ Exemple d'utilisation des netgroups

```
root@192.168.90.2# cat /etc/netgroup (sur le serveur NIS)
mes_mach (192.168.69.1,,grid5000) (192.168.69.2,,grid5000)
net_admins (,ogluck,) (,root,)
mes_users (,toto,) (,titi,)
root@192.168.90.1# cat /etc/exports (sur le serveur NFS)
#j'autorise mes_machines à monter /home...
/home @mes_mach(rw,root_squash,async)
root@192.168.69.2# tail -2 /etc/passwd (sur le client NIS)
#lignes toto,titi de la map passwd non visibles du client
-@mes_users:
#lignes root,ogluck de la map passwd visibles du client
+@net_admins:::

```

■ Pour installer les NIS (mise en place de la base...)

- `ypinit -m` sur le serveur NIS maître
- `ypinit -s master-server` sur serveur esclave

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 111

111

NIS : en pratique...

■ Savoir si les services sont en place

```
root@ 192.168.69.2# rpcinfo -u 192.168.90.2 ypserv
program 100004 version 1 ready and waiting
program 100004 version 2 ready and waiting
root@ 192.168.69.2# rpcinfo -u 192.168.69.2 ypbind
program 100007 version 1 ready and waiting
program 100007 version 2 ready and waiting
root@ 192.168.69.2# yppwhich
192.168.90.2
```

■ Contrôle de l'accès au serveur NIS

```
root@ 192.168.90.2# cat /etc/ypserv.securenets
#This file defines the access rights to your NIS server
255.0.0.0 127.0.0.0
255.255.255.0 192.168.69.0
```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 112

112

NIS : les procédures distantes

■ Exemples de RPC permettant l'interrogation d'un serveur NIS

- `YPPROC_DOMAIN` : retourne vrai si le serveur répond au domaine reçu en paramètre, faux sinon ; utilisé en particulier par `ypbind` pour savoir si le serveur NIS actuel répond toujours (~ 20 secondes) sinon il en cherche un autre dans `/etc/yp.conf`
- `YPPROC_MATCH` : retourne la valeur associée à la clé passée en paramètre (`ypmatch clé mapname`)
- `YPPROC_ALL` : retourne tous les couples clé/valeur de la map passée en paramètre
- `YPPROC_MAPLIST` : retourne la liste de tous les noms de map pour le domaine passé en paramètre
- `YPPROC_XFR` : permet de demander au serveur la mise à jour d'une map (dans la crontab, sur les serveurs esclave : `ypxfr passwd.byname`)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 113

113

NIS : évolutions

■ Défauts des NIS

- pas d'authentification des clients NIS : il suffit de connaître le nom de domaine pour interroger le serveur et connaître le contenu de ses maps
- les maps sont transmises en totalité même en cas de faible modification de leurs contenus
- pas adapté aux WAN (*broadcast...*)
- NIS+ un successeur éphémère sans succès qui a été officiellement abandonné au profit de LDAP
- Cependant, les NIS sont encore largement utilisés dans le cadre d'un réseau local simple

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 114

114

Partie 2 : Applications de l'Internet de type Client/Serveur (suite1)

Olivier GLÜCK
Université LYON 1/UFR d'Informatique
Olivier.Gluck@ens-lyon.fr
<http://www710.univ-lyon1.fr/~ogluck>



115

Plan de la partie 2

- Introduction / Rappel
- Connexions à distance (telnet/rlogin/rsh/ssh/X11)
- Applications de transfert de fichiers (FTP/TFTP)
- Accès aux fichiers distants (NFS/SMB)
- Gestion d'utilisateurs distants (NIS)
- **DNS : un annuaire distribué**
- LDAP : un annuaire fédérateur sécurisé
- La messagerie électronique (SMTP/POP/IMAP)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

116

DNS : un annuaire distribué des adresses de l'Internet

Le système DNS
Une base de données distribuée
Notions de zones et domaines
Les différents types de serveurs
Résolutions récursives et itératives
Cache DNS, Format des messages DNS
Commandes et fichiers liés au DNS



117

DNS : Domain Name System

- Gens : plusieurs identifiants
 - #sécu, nom, #Passeport
- Hôtes, routeurs :
 - adresse IP (32 bits)
 - "nom" :
 - www.google.com
 - www.education.gouv.fr
- Problème résolu par le DNS : Comment relier les adresses IP utilisées pour acheminer les paquets aux noms utilisés par les utilisateurs ou les applications ?

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

118

DNS : Domain Name System

- C'est une base de données **distribuée**
 - implantée dans une hiérarchie de serveurs de noms
- C'est un protocole applicatif
 - les hôtes, routeurs, serveurs de noms communiquent pour effectuer la traduction
 - DNS est utilisé par d'autres protocoles applicatifs mais n'est pas utilisé directement par l'application comme SMTP...
 - modèle Client/Serveur : un émetteur interroge un serveur de noms (serveur DNS)
 - port 53/UDP (ou 53/TCP pour les mises à jour)
 - RFC 1034, 1035, 2181, ...

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

119

119

Les services fournis par le DNS

- Le service principal : la traduction d'adresses
- Autres services :
 - permettre le "*Host aliasing*" : donner un pseudonyme à une machine qui a un nom peu parlant
 - permettre le "*Mail server aliasing*" : un serveur Web et un serveur Mail peuvent avoir le même pseudonyme même s'ils n'ont pas la même adresse IP (2 machines ≠)
 - permettre la répartition de la charge : un nom de serveur Web ou Mail peut correspondre à plusieurs adresses IP (serveurs Web ou Mail répliqués) avec un système de rotation dans les réponses du serveur DNS
- Pour l'utilisateur, le DNS n'est qu'une boîte noire mais en réalité très compliquée
 - une requête DNS peut impliquer plusieurs serveurs de noms répartis dans le monde entier

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

120

120

Un système centralisé ?

- Pourquoi pas de DNS centralisé ? Un seul serveur contiendrait toutes les correspondances requises par les applications de l'Internet
 - dimension de l'Internet : trop de correspondances à gérer, nombre de requêtes au serveur trop important
 - tolérance aux pannes : si le serveur DNS tombe, tout l'Internet aussi !
 - volume de trafic impossible à supporter par un seul serveur
 - délais de réponse : il faut faire en sorte que la réponse soit la plus proche possible du demandeur
 - problème lié à la maintenance et aux mises à jour perpétuelles de la base

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

121

121

Un système distribué

- Aucun serveur ne connaît toutes les correspondances nom <--> adresse IP
 - si un serveur ne connaît pas une correspondance, il interroge un autre serveur jusqu'à atteindre le serveur détenant l'information désirée
- Trois types de serveur DNS
 - les serveurs de noms locaux (à qui s'adressent les requêtes locales ; en charge de la résolution)
 - les serveurs de noms racine (sont censés savoir comment s'approcher de la réponse)
 - les serveurs de noms de source autorisée (contiennent les correspondances "officielles")

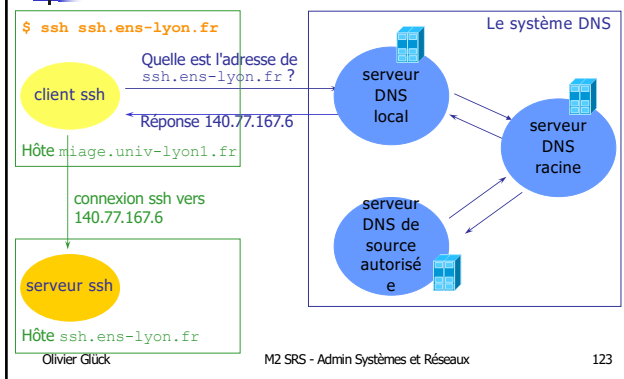
Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

122

122

Un système distribué

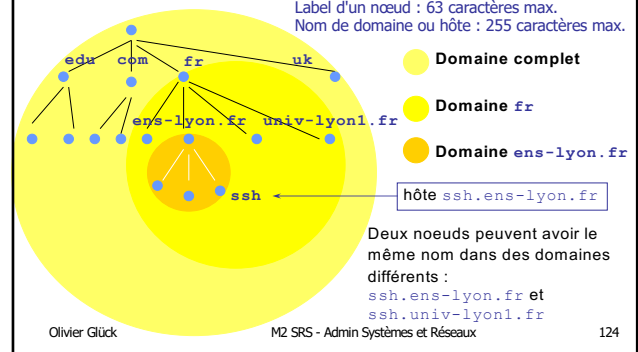


123

La notion de domaine DNS

Un domaine est un sous-arbre entier de l'espace de nommage

Label d'un nœud : 63 caractères max.
Nom de domaine ou hôte : 255 caractères max.



124

La notion de domaine DNS

- Le premier niveau de l'arbre
 - Top Level Domain (TLD)
 - géré par l'ICANN (*Internet Corporation for Assigned Names and Numbers*)
 - on distingue deux catégories de TLD
 - les "Generic TLD" : .com, .org, .gov, .gouv, .net, ...
 - les "Countries TLD" : .fr, .uk, .us, .jp, ... (240 en tout)
- La gestion des autres niveaux est laissée aux entités correspondantes (AFNIC pour .fr)
 - zone DNS : un sous-arbre de l'arbre administré séparément par un organisme qui gère la délégation des noms et sous-domaines de la zone

Olivier Glück

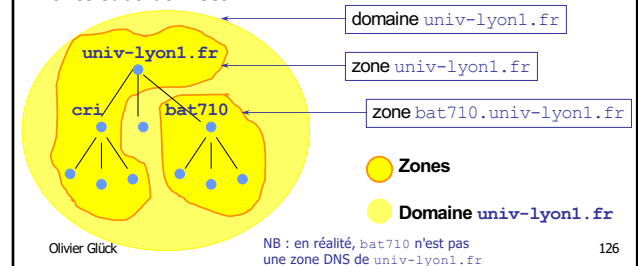
M2 SRS - Admin Systèmes et Réseaux

125

125

La notion de zone DNS

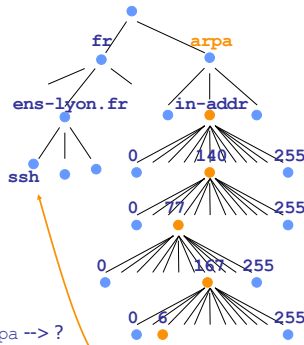
- Une zone = une administration centralisée avec au moins un serveur DNS (généralement 1 primaire et 1 secondaire)
 - le secondaire (redondance) met à jour ses données à partir du primaire
- Une zone doit connaître les adresses des serveurs DNS des zones subordonnées



126

La résolution de noms inverse

- Retrouver le nom canonique à partir de l'adresse IP
- Le domaine `arpa` : un domaine particulier géré par l'ICANN permettant la résolution inverse
- Résolution
`ssh.ens-lyon.fr --> ?`
- Résolution inverse
`6.167.77.140.in-addr.arpa --> ?`



127

Les différents types de serveur DNS

- Les serveurs de noms locaux
 - chaque organisation a un serveur de noms local
 - serveur DNS par défaut de la zone
 - contient parfois les correspondances relatives à la zone de l'organisation
 - toutes les requêtes DNS en provenance de cette organisation vont vers ce serveur de nom local
- Les serveurs de noms racine [RFC 2870]
 - il existe actuellement 13 serveurs racine dans l'Internet (liste sur <http://gnso.icann.org/gtld-registries/>)
 - chaque serveur DNS local connaît un serveur de noms racine qu'il peut interroger s'il ne connaît pas une correspondance
 - un serveur de noms racine connaît au moins les serveurs de source autorisée du premier niveau (`.fr.`, ...)

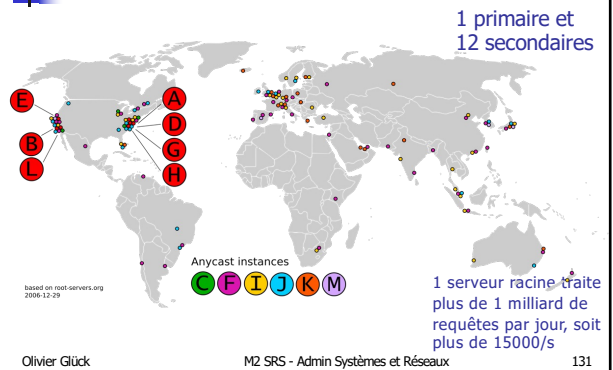
128

Les différents types de serveur DNS

- Un serveur de noms racine qui ne connaît pas une correspondance interroge un autre serveur de noms le rapprochant de la réponse, généralement le serveur de noms de source autorisée qui connaît la correspondance
- Les serveurs de noms de source autorisée
 - chaque hôte est enregistré auprès d'au moins deux "authoritative servers" (le primaire et le secondaire), qui stocke son adresse IP et son nom
 - un serveur de noms est dit de source autorisée pour un hôte s'il est responsable de la correspondance nom/@ pour cet hôte (serveur primaire de la zone)
 - un serveur de noms local n'est pas forcément de source autorisée (ex. `bat710.univ-lyon1.fr`)

129

Les différents types de serveur DNS



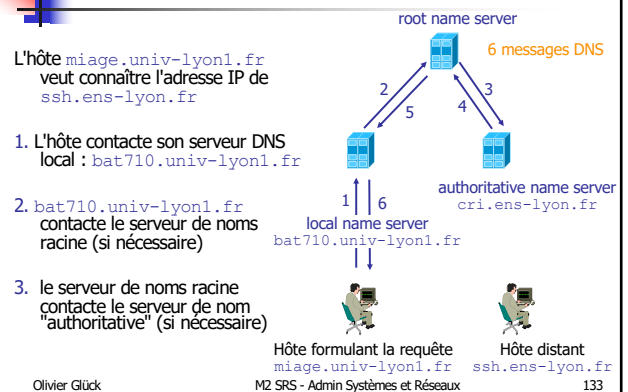
131

Résolution de noms récursive/itérative

- Requête récursive
 - la machine qui demande la résolution de nom contacte un serveur DNS et attend que ce dernier lui retourne la réponse désirée
- Requête itérative
 - le serveur de noms contacté fournit en réponse le nom d'un autre serveur DNS à contacter pour avancer dans la résolution
 - "je ne connais pas ce nom mais demande à ce serveur"
- Dans le cheminement d'une résolution de nom, certaines requêtes peuvent être itératives, d'autres récursives

132

Principe d'une résolution récursive



133

Principe d'une résolution récursive

Cas où un serveur intermédiaire est nécessaire

8 messages DNS

- Le serveur de noms racine ne connaît pas forcément le serveur de source autorisée qui le renseignera sur la correspondance recherchée
- Passage par un ou plusieurs serveurs de noms intermédiaires

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 134

134

Principe d'une résolution itérative

A qui dois-je demander pour ssh.ens-lyon.fr ?

Requête/Réponse itératives

- Généralement les requêtes sont toutes récursives sauf celle entre le serveur de noms local et le serveur de noms racine
- > permet de moins solliciter le serveur racine (moins exigeant en termes de ressources)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 135

135

La mise en mémoire cache DNS

- Idee générale des caches : réduire le temps de réponse
 - ici, le temps de réponse d'une résolution de nom, en diminuant le nombre de messages DNS en transit nécessaires !
 - le serveur de noms (quelconque) stocke dans son cache les informations récentes (en particulier les enregistrements de type NS)
 - comme la mémoire n'est pas infinie et que les données du cache peuvent ne plus être valables au bout d'un certain temps, les données expirent du cache après un certain temps TTL (environ 2 jours)
 - un serveur qui mémorise dans son cache un enregistrement DNS n'a pas autorité dessus --> spécifie "no authoritative" dans la réponse

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 136

136

Etude de différents cas de figure

Source : http://media.pearsoncmg.com/aw/aw_kurose_network_2/applets/dns/dns.html

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 137

137

Les messages DNS [RFC 1034, 1035]

- Un message de réponse DNS contient un ou plusieurs RR (*Resource Record*)
 - l'unité de stockage d'une correspondance dans le cache RR = (Nom, Type, Classe, TTL, Valeur)
 - Type (16 bits) : type de l'enregistrement ; la signification de Nom (N octets) et Valeur dépend du type
 - Classe (16 bits) : famille de protocoles ; Classe=1 pour Internet (IN)
 - TTL (32 bits) : temps en secondes pendant lequel il faut conserver l'enregistrement dans le cache
 - Valeur = RDLENGTH (16 bits) + RDATA (N octets)
- Un message de requête DNS contient des questions du type (Nom, Type, Classe)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 138

138

Les messages DNS [RFC 1034, 1035]

- Un unique format pour les Requêtes/Réponses
- 12 octets d'en-tête DNS
 - un identifiant qui permet au client d'associer la réception d'une réponse à une requête formulée
 - les fanions : 2 octets

```

+-----+-----+-----+-----+-----+-----+-----+-----+
| Q|R|  Opcode  |&&|TC|RD|RA|  Z  | RCODE |
+-----+-----+-----+-----+-----+-----+-----+
QR : 0 si requête, 1 si réponse
OPCODE (4 bits) : 0 (QUERY), 1 (Inverse QUERY), ...
AA : 1 si le serveur de noms qui répond a autorité sur la réponse sollicitée
TC : le message a été tronqué --> utiliser TCP
RD : 1 dans la requête si le client souhaite que le serveur de noms effectue
      une récursion s'il ne dispose pas de la traduction demandée
RA : 1 dans la réponse si le serveur de noms contacté assure la récursion
AD et CD (dans Z) : "Authentic Data" et "Checking Disabled" (DNSSEC)
RCODE (4 bits) : 0 (pas d'erreur), 3 (Name error), 5 (Refused), ...
  
```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 139

139

Les messages DNS [RFC 1034, 1035]

512 octets max. avec UDP

Identification	Fanions	} 12 octets d'en-tête
Nombre de questions	Nombre de réponses (RR)	
Nombre de "authority servers"	Nombre de RR supplémentaires	} QUESTION SECTION
Questions (nombre variable de questions)		
Réponses (RRs répondant à la demande)		ANSWER SECTION
Noms des serveurs de source autorisée (RRs des serveurs primaire/secondaire)		AUTHORITY SECTION
Adresses des serveurs de source autorisée (nb. variable de RRs)		ADDITIONAL SECTION

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 140

140

Les messages DNS [RFC 1034, 1035]

```

ogluck@lima:~$ host -a ssh.ens-lyon.fr
Trying 'ssh.ens-lyon.fr'
:: ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 11431
:: Flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 3, ADDITIONAL: 3
} 12 octets d'en-tête
:: QUESTION SECTION:
:ssh.ens-lyon.fr.
      IN      ANY
:: ANSWER SECTION:
ssh.ens-lyon.fr. 7200 IN CNAME fulmar.ens-lyon.fr.
} Serveur primaire
:: AUTHORITY SECTION:
ens-lyon.fr. 7200 IN NS cri.ens-lyon.fr.
ens-lyon.fr. 7200 IN NS imag.ens-lyon.fr.
ens-lyon.fr. 7200 IN NS ens.ens-lyon.fr.
} Serveurs secondaires
:: ADDITIONAL SECTION:
cri.ens-lyon.fr. 7200 IN A 140.77.1.32
imag.ens-lyon.fr. 172857 IN A 129.88.30.1
ens.ens-lyon.fr. 7200 IN A 140.77.1.183
Received 162 bytes from 140.77.1.32#53 in 0 ms
ogluck@lima:~$
  
```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 141

141

Les messages DNS [RFC 1034, 1035]

- **Type=A** (val=1) : sert à décrire une correspondance
Nom=nom d'hôte (canonique), Value=@IPv4
- **Type=AAAA** (val=28, RFC 1886) : idem mais adresse IPv6
Nom=nom d'hôte, Value=@IPv6
- **Type=PTR** (val=12) : sert à la résolution inverse
Nom=un nom de la zone arpa, Value=nom canonique (valeur pointée)
- **Type=NS** (val=2) : sert à associer un nom de domaine à un serveur de noms de source autorisée
Nom=domaine, Value=nom du serveur de noms
- **Type=CNAME** (val=5) : sert à définir un alias pour un hôte
Nom=un alias, Value=nom canonique (le vrai nom)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 142

142

Les messages DNS [RFC 1034, 1035]

- **Type=MX** (val=15) : alias réservés aux serveurs mail permettant d'associer plusieurs serveurs de mail avec différentes priorités à une même adresse (RFC 974)
Nom=un alias, Value=nom canonique d'un serveur de mail
- **Type=SOA** (val=6) : sert à donner des infos sur la zone gérée
Nom=nom d'une zone, Value=informations sur la zone
- **Type=ANY** (val=255) : utilisé dans les questions pour indiquer n'importe quel type (*)
- **Type=AXFR** (val=252) : utilisé dans les questions pour demander le transfert d'une zone entière (mise à jour d'un serveur secondaire...)
- **Type=HINFO** (val=13) : sert à indiquer les CPU et OS de l'hôte interrogé

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 143

143

Les messages DNS [RFC 1034, 1035]

- Exemples

```

ssh.ens-lyon.fr. CNAME fulmar.ens-lyon.fr.
ens-lyon.fr. NS cri.ens-lyon.fr.
ens-lyon.fr. NS ens.ens-lyon.fr.
cri.ens-lyon.fr. A 140.77.1.32
relaismtp.ens-lyon.fr. CNAME pluvier.ens-lyon.fr.
ens-lyon.fr. MX 20 pluvier.ens-lyon.fr.
ens-lyon.fr. MX 30 pluvier2.ens-lyon.fr.
listes.ens-lyon.fr. MX 20 pluvier.ens-lyon.fr.
fulmar.ens-lyon.fr. A 140.77.167.6
6.167.77.140.in-addr.arpa.PTR fulmar.ens-lyon.fr
  
```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 144

144

Routage de courrier et DNS [RFC 974]

- Centraliser la réception des messages sur une machine qui a un système plus robuste
 - anti-virus, anti-spam, ...
 - seule machine accessible sur le port 25 depuis l'extérieur via le pare-feu
- Les MX permettent ensuite de répartir la charge sur différents serveurs de mail et de disposer de serveurs de secours
 - en cas de saturation, le serveur SMTP peut aiguiller les messages via un autre serveur SMTP interne

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 145

145

La commande host

```

xterm
ogluck@lima:~$ host
Usage: host [-s<dir/fw>] [-c class] [-n] [-N ndots] [-t type] [-W time]
          [-R number] hostname [server]
-a is equivalent to -v -t *
-c specifies query class for non-IN data
-C compares SOA records on authoritative nameservers
-d is equivalent to -v
-l lists all hosts in a domain, using A/R/R
-n Use the nibble form of IPv6 reverse lookup
-N changes the number of dots allowed before root lookup is done
-r disables recursive processing
-R specifies number of retries for UDP packets
-t specifies the query type
-T enables TOP/1P mode
-v enables verbose output
-w specifies to wait forever for a reply
-W specifies how long to wait for a reply
ogluck@lima:~$
    
```

```

ogluck@lima:~$ host ssh.ens-lyon.fr
ssh.ens-lyon.fr is an alias for fulmar.ens-lyon.fr.
fulmar.ens-lyon.fr has address 140.77.167.6
    
```

146

La commande host

```

xterm
ogluck@lima:~$ host -l ens-lyon.fr | grep pluvier
ens-lyon.fr mail is handled by 20 pluvier.ens-lyon.fr.
relaismtpl.ens-lyon.fr is an alias for pluvier.ens-lyon.fr.
listes.ens-lyon.fr mail is handled by 20 pluvier.ens-lyon.fr.
pluvier.ens-lyon.fr has address 140.77.167.5
psmn.ens-lyon.fr mail is handled by 20 pluvier.ens-lyon.fr.
ump.ens-lyon.fr mail is handled by 20 pluvier.ens-lyon.fr.
ogluck@lima:~$ host -a -l ens-lyon.fr | grep pluvier
ens-lyon.fr. 7200 IN MX 20 pluvier.ens-lyon.fr.
relaismtpl.ens-lyon.fr. 7200 IN CNAME pluvier.ens-lyon.fr.
listes.ens-lyon.fr. 7200 IN MX 20 pluvier.ens-lyon.fr.
pluvier.ens-lyon.fr. 7200 IN A 140.77.167.5
psmn.ens-lyon.fr. 7200 IN MX 20 pluvier.ens-lyon.fr.
ump.ens-lyon.fr. 7200 IN MX 20 pluvier.ens-lyon.fr.
ogluck@lima:~$
    
```

- Pour connaître les serveurs de source autorisée d'une zone :
host -a nom_zone OU **host -t ns nom_zone**

147

Les messages DNS - type PTR

6.167.77.140.in-addr.arpa est un pointeur vers fulmar.ens-lyon.fr

```

xterm
ogluck@lima:~$ host 140.77.167.6
6.167.77.140.in-addr.arpa domain name pointer fulmar.ens-lyon.fr.
ogluck@lima:~$ host -a 140.77.167.6
Trying "6.167.77.140.in-addr.arpa"
:: -->HEADER<<- opcode: QUERY, status: NOERROR, id: 5334
:: flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 3, ADDITIONAL: 3
:: QUESTION SECTION:
;6.167.77.140.in-addr.arpa. IN PTR
:: ANSWER SECTION:
6.167.77.140.in-addr.arpa. 7200 IN PTR fulmar.ens-lyon.fr.
:: AUTHORITY SECTION:
77.140.in-addr.arpa. 7200 IN NS cri.ens-lyon.fr.
77.140.in-addr.arpa. 7200 IN NS copntc3.in2p3.fr.
77.140.in-addr.arpa. 7200 IN NS ens.ens-lyon.fr.
:: ADDITIONAL SECTION:
cri.ens-lyon.fr. 7200 IN A 140.77.1.32
copntc3.in2p3.fr. 7200 IN A 134.158.69.191
ens.ens-lyon.fr. 7200 IN A 140.77.1.183
Received 187 bytes from 140.77.1.32#53 in 1 ms
ogluck@lima:~$
    
```

148

Les messages DNS - types AXFR, SOA

Demande du contenu de toute la zone ens-lyon.fr

```

xterm
ogluck@lima:~$ host -a -l ens-lyon.fr | head
Trying "ens-lyon.fr"
:: -->HEADER<<- opcode: QUERY, status: NOERROR, id: 30097
:: flags: qr aa; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
:: QUESTION SECTION:
;ens-lyon.fr. IN AXFR
:: ANSWER SECTION:
ens-lyon.fr. 7200 IN SOA cri.ens-lyon.fr.
admin.ens-lyon.fr. 2004032305 21600 3600 3600000 7200
ogluck@lima:~$
    
```

Indique l'email de la personne responsable de la zone (lire admin@)
 Numéro de série de la zone
 TTL exporté dans les RR de la zone
 Serveur primaire

149

Le DNS Round-robin

- Les alias : plusieurs noms correspondent à une même adresse IP
- Le Round-robin : à un même nom correspond plusieurs adresses IP --> permet d'avoir de la redondance (plusieurs RRs de type A)
- Le DNS change l'ordre à chaque nouvelle requête pour répartir la charge

```

xterm
ogluck@lima:~$ host -t a univ-lyon1.fr
univ-lyon1.fr has address 134.214.139.200
univ-lyon1.fr has address 193.48.219.3
univ-lyon1.fr has address 195.48.219.4
univ-lyon1.fr has address 195.220.110.100
univ-lyon1.fr has address 134.214.9.100
univ-lyon1.fr has address 134.214.28.2
ogluck@lima:~$ host -t a univ-lyon1.fr
univ-lyon1.fr has address 193.48.219.3
univ-lyon1.fr has address 195.48.219.4
univ-lyon1.fr has address 195.220.110.100
univ-lyon1.fr has address 134.214.9.100
univ-lyon1.fr has address 134.214.28.2
univ-lyon1.fr has address 134.214.139.200
ogluck@lima:~$
    
```

150

La mise en mémoire cache

```

ogluck/home/ogluck/proxy710
##### host -a bat710
Trying "bat710.univ-lyon1.fr"
:: -->HEADER<<- opcode: QUERY, status: NOERROR, id: 36701
:: flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 3, ADDITIONAL: 3
:: QUESTION SECTION:
;bat710.univ-lyon1.fr. IN ANY
:: ANSWER SECTION:
bat710.univ-lyon1.fr. 375878 IN A 134.214.88.10
:: AUTHORITY SECTION:
univ-lyon1.fr. 326164 IN NS dns.univ-lyon1.fr.
univ-lyon1.fr. 326164 IN NS dns2.univ-lyon1.fr.
univ-lyon1.fr. 326164 IN NS dns1.univ-lyon1.fr.
:: ADDITIONAL SECTION:
dns.univ-lyon1.fr. 326164 IN A 134.214.100.6
dns2.univ-lyon1.fr. 326164 IN A 134.214.100.245
dns1.univ-lyon1.fr. 326164 IN A 134.214.100.3
Received 158 bytes from 127.0.0.1#53 in 1 ms
##### host -a bat710 | grep "bat710.univ-lyon1.fr"
bat710.univ-lyon1.fr. 375871 IN A 134.214.88.10
##### host -a bat710 | grep "bat710.univ-lyon1.fr"
bat710.univ-lyon1.fr. 375864 IN A 134.214.88.10
##### host -a bat710 | grep "bat710.univ-lyon1.fr"
bat710.univ-lyon1.fr. 375862 IN A 134.214.88.10
##### host -a bat710 | grep "bat710.univ-lyon1.fr"
bat710.univ-lyon1.fr. 375859 IN A 134.214.88.10
#####
    
```

Réponse non autoritative : flag aa absent (réponse provenant du cache)

Lors de chaque nouvelle requête, le TTL a diminué !

151

La commande dig

- Rend les mêmes services que `host` mais est encore plus bas niveau : permet en particulier de voir l'ensemble des requêtes/réponses

```

oglu@home/oglu@proxy710
#####$ hostname
proxy710
#####$ dig +trace ssh.ens-lyon.fr

;<> DiG 9.2.1 <> +trace ssh.ens-lyon.fr
;; global options: printed
.      120835 IN NS      I.ROOT-SERVERS.NET.
.      120835 IN NS      J.ROOT-SERVERS.NET.
.      120835 IN NS      K.ROOT-SERVERS.NET.
.      120835 IN NS      L.ROOT-SERVERS.NET.
.      120835 IN NS      M.ROOT-SERVERS.NET.
.      120835 IN NS      A.ROOT-SERVERS.NET.
.      120835 IN NS      B.ROOT-SERVERS.NET.
.      120835 IN NS      C.ROOT-SERVERS.NET.
.      120835 IN NS      D.ROOT-SERVERS.NET.
.      120835 IN NS      E.ROOT-SERVERS.NET.
.      120835 IN NS      F.ROOT-SERVERS.NET.
.      120835 IN NS      G.ROOT-SERVERS.NET.
.      120835 IN NS      H.ROOT-SERVERS.NET.
;; Received 436 bytes from 127.0.0.1#53(127.0.0.1) in 15 ms
    
```

Un serveur DNS local donne les serveurs de source autorisée pour la zone ". " (serveurs racines)

152

La commande dig

- Suite... Le serveur racine I donne les serveurs de source autorisée pour la zone ".fr."

```

Fr.      172800 IN NS      DNS.INRIA.fr.
Fr.      172800 IN NS      DNS.PRINCETON.EDU.
Fr.      172800 IN NS      NS1.NIC.fr.
Fr.      172800 IN NS      NS2.NIC.fr.
Fr.      172800 IN NS      NS3.NIC.fr.
Fr.      172800 IN NS      NS5.DOMAIN-REGISTRY.NL.
Fr.      172800 IN NS      NS-EXT.VIX.COM.
Fr.      172800 IN NS      DNS.ES.WISC.EDU.
;; Received 364 bytes from 192.36.148.17#53(I.ROOT-SERVERS.NET) in 56 ms
ens-lyon.fr. 345600 IN NS      imag.imag.fr.
ens-lyon.fr. 345600 IN NS      cri.ens-lyon.fr.
ens-lyon.fr. 345600 IN NS      ens.ens-lyon.fr.
;; Received 141 bytes from 193.51.208.13#53(DNS.INRIA.fr) in 15 ms
ssh.ens-lyon.fr. 7200 IN CNAME fulmar.ens-lyon.fr.
fulmar.ens-lyon.fr. 7200 IN A      140.77.167.6
ens-lyon.fr. 7200 IN NS      ens.ens-lyon.fr.
ens-lyon.fr. 7200 IN NS      imag.imag.fr.
ens-lyon.fr. 7200 IN NS      cri.ens-lyon.fr.
;; Received 178 bytes from 129.88.30.1#53(imag.imag.fr) in 7 ms
#####$
    
```

Finalement, la réponse est donnée par `imag.imag.fr`

153

Du côté client... - le resolver

- Le *resolver* a en charge les résolutions de noms (inverse ou pas) chaque fois que cela est nécessaire --> `man resolver`
- Deux fichiers de configuration lui sont associés
 - `/etc/resolv.conf` permet de paramétrer les requêtes DNS effectuées (`man resolv.conf`)
 - `/etc/host.conf` permet de paramétrer le *resolver* (`man host.conf`), en particulier ordre de résolution `order hosts,bind,nis`
 - `/etc/nsswitch.conf` est consulté en premier s'il existe
- Extrait de l'API du *resolver* pour les applications
 - `gethostbyname(name)`
 - `gethostbyaddr(addr)`

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 154

154

Du côté client... - les commandes

```

oglu@lima:~$ cat /etc/hostname
lima
oglu@lima:~$ dnsdomainname
cri2000.ens-lyon.fr
oglu@lima:~$ hostname
lima
oglu@lima:~$ hostname --fqdn
lima.cri2000.ens-lyon.fr
oglu@lima:~$ head -2 /etc/hosts
127.0.0.1      localhost
140.77.13.131  lima.cri2000.ens-lyon.fr lima titi
oglu@lima:~$ ping titi
PING lima.cri2000.ens-lyon.fr (140.77.13.131): 56 data bytes
64 bytes from 140.77.13.131: icmp_seq=0 ttl=0 time=0.0 ms

--- lima.cri2000.ens-lyon.fr ping statistics ---
1 packets transmitted, 1 packets received, 0% packet loss
oglu@lima:~$ hostname -i
140.77.13.131
oglu@lima:~$
    
```

Le nom complet de la machine est fixé au démarrage (souvent nom court dans `/etc/hostname`)

Nom de domaine

Nom court (avant le premier .) Permet à `root` de changer le nom

Nom complet

`/etc/hosts` permet de définir des alias (@IP nom_canonique aliases)

Résolution inverse par le *resolver*

155

Le fichier /etc/nsswitch.conf

Permet de spécifier l'ordre des méthodes de résolutions (ligne `hosts` pour la résolution de noms) `man nsswitch.conf`

```

lima-export/home/oglu
lima-export/home/oglu#cat /etc/nsswitch.conf
# /etc/nsswitch.conf
#
# Example configuration for GNU Name Service Switch functionality.
# Information about this file is available in the 'libc-doc' package.

passwd:      files
group:       files
shadow:      files

hosts:       files nis dns
networks:    files

protocols:   db files
services:    db files
ethers:      db files
rpc:         db files

netgroup:    db files nis
lima-export/home/oglu#
    
```

Ici `/etc/hosts`, `map hosts.by...` via les NIS, DNS

Pour chaque source, on peut préciser l'action à entreprendre selon le statut retourné ; par défaut : [SUCCESS=return NOTFOUND=continue UNAVAIL=continue TRYAGAIN=forever]

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 156

156

Configuration d'un poste de travail

Propriétés de Protocole Internet (TCP/IP)

Paramètres TCP/IP avancés

Général

Les paramètres IP peuvent être déterminés automatiquement si votre réseau le permet. Sinon, vous devez demander les paramètres IP appropriés à votre administrateur réseau.

Obtenir une adresse IP automatiquement

Utiliser l'adresse IP suivante:

Adresse IP: 134.214.91.36

Masque de sous-réseau: 255.255.252.0

Passerelle par défaut: 134.214.88.1

Obtenir les adresses des serveurs DNS automatiquement

Utiliser l'adresse de serveur DNS suivante:

Serveur DNS préféré: 134.214.88.10

Serveur DNS auxiliaire:

Avancé...

OK Annuler

Paramètres DNS

Adresses des serveurs DNS, dans l'ordre d'utilisation:

134.214.88.10

Les trois paramètres suivants sont appliqués à toutes les connexions pour lesquelles TCP/IP est activé. Pour la résolution des noms non qualifiés:

Ajouter des suffixes DNS principaux et spécifiques aux connexions

Ajouter des suffixes parents du suffixe DNS principal

Ajouter ces suffixes DNS (dans l'ordre):

Suffixe DNS pour cette connexion: `lima-lyon1.fr`

Enregistrer les adresses de cette connexion dans le système DNS

Utiliser le suffixe DNS de cette connexion pour l'enregistrement DNS

Indiquer le(s) serveur(s) de noms locaux

Suffixe DNS principal pour cette connexion

157

Le fichier /etc/resolv.conf

```
ogluck@lima:~$ cat /etc/resolv.conf
search univ-lyon1.fr
search cri2000.ens-lyon.fr
search ens-lyon.fr
search abblol.org
nameserver 140.77.1.32
nameserver 140.77.1.183
```

```
ogluck@lima:~$ nslookup univ-lyon1.fr
Server: 140.77.1.183
Address: 140.77.1.183
31816 IN IN ANY
2004032600 21600 3600 3600000 86400
71801 IN NS DNS_CS.WISSE.EDU.
71801 IN NS ns1.nic.fr.
71801 IN NS NS2.nic.fr.
71801 IN NS DNS_INRIA.fr.
71801 IN NS NS2.nic.fr.
71801 IN NS DNS_PRINCETON.EDU.
71801 IN NS NS-EXT.VIX.COM.
71801 IN NS NS3.DOMAIN-REGISTRY.NL.
67200 IN A 128.106.2.10
182002 IN A 192.33.0.1
165949 IN AAAA 2001:680:2006::1:11
244465 IN A 192.156.0.48
4295 IN A 193.51.200.13
182160 IN A 192.33.0.4
133084 IN A 128.112.129.15
153747 IN A 204.152.194.24
255 IN AAAA 2001:480:0002::12
153.176.144.8
```

Le côté client du DNS : permet de spécifier comment et à qui formuler les requêtes

- Au maximum 6 lignes *search*
- Au maximum 3 lignes *nameserver*

Olivier Glück 158

158

Le fichier /etc/resolv.conf

■ Importance de l'ordre des suffixes de recherche

```
ogluck@lima:~$ nslookup -a miage
Trying 'miage.univ-lyon1.fr.'
Server: 140.77.1.183
Address: 140.77.1.183
31816 IN IN ANY
2004032600 21600 3600 3600000 86400
71801 IN NS DNS_CS.WISSE.EDU.
71801 IN NS ns1.nic.fr.
71801 IN NS NS2.nic.fr.
71801 IN NS DNS_INRIA.fr.
71801 IN NS NS2.nic.fr.
71801 IN NS DNS_PRINCETON.EDU.
71801 IN NS NS-EXT.VIX.COM.
71801 IN NS NS3.DOMAIN-REGISTRY.NL.
67200 IN A 128.106.2.10
182002 IN A 192.33.0.1
165949 IN AAAA 2001:680:2006::1:11
244465 IN A 192.156.0.48
4295 IN A 193.51.200.13
182160 IN A 192.33.0.4
133084 IN A 128.112.129.15
153747 IN A 204.152.194.24
255 IN AAAA 2001:480:0002::12
153.176.144.8
```

```
ogluck@lima:~$ nslookup -a miage cri2000.ens-lyon.fr
Trying 'miage.cri2000.ens-lyon.fr.'
Server: 140.77.1.183
Address: 140.77.1.183
31816 IN IN ANY
2004032600 21600 3600 3600000 86400
71801 IN NS DNS_CS.WISSE.EDU.
71801 IN NS ns1.nic.fr.
71801 IN NS NS2.nic.fr.
71801 IN NS DNS_INRIA.fr.
71801 IN NS NS2.nic.fr.
71801 IN NS DNS_PRINCETON.EDU.
71801 IN NS NS-EXT.VIX.COM.
71801 IN NS NS3.DOMAIN-REGISTRY.NL.
67200 IN A 128.106.2.10
182002 IN A 192.33.0.1
165949 IN AAAA 2001:680:2006::1:11
244465 IN A 192.156.0.48
4295 IN A 193.51.200.13
182160 IN A 192.33.0.4
133084 IN A 128.112.129.15
153747 IN A 204.152.194.24
255 IN AAAA 2001:480:0002::12
153.176.144.8
```

Received 93 bytes from 140.77.1.32#53 in 1 ms

Received 93 bytes from 140.77.1.32#53 in 0 ms

159

159

Le serveur de noms named (BIND)

- BIND : Berkeley Internet Name Domain
 - <http://www.isc.org/sw/bind>
 - implantation d'un serveur DNS du domaine publique
- Le démon répondant aux requêtes DNS est named
 - fichier de configuration : *named.conf*
 - il faut y associer les fichiers décrivant les zones administrées (syntaxe master files : voir RFC 1035)
 - > dans */etc/namedb* ou */etc/bind*
- Des utilitaires
 - rndc* permet de contrôler à distance le fonctionnement de named (avec authentification)
 - named-checkconf* et *named-checkzone* permettent de vérifier la syntaxe des fichiers de zones ou config.

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 160

160

Le fichier named.conf

```
man named.conf
#####$ cat /etc/bind/named.conf // be authoritative for the local host
// This is the primary configuration file // forward and reverse zones, and
// for the BIND DNS server named. // for broadcast zones as per RFC 1912
options {
    // répertoire de travail de named
    directory "/var/cache/bind";
    // si le serveur n'a pas la réponse
    // il forward à un autre
    forward first;
    forwarders {
        134.214.88.23;
        134.214.88.10;
    };
};
// prime the server with knowledge
// of the root servers
zone "." {
    // copie m.a.j. au démarrage
    type hint;
    file "/etc/bind/db.root";
};
zone "localhost" {
    type master; // autorité pour la zone
    file "/etc/bind/db.localhost";
};
zone "127.in-addr.arpa" {
    type master;
    file "/etc/bind/db.127";
};
zone "0.in-addr.arpa" {
    type master;
    file "/etc/bind/db.0";
};
zone "255.in-addr.arpa" {
    type master;
    file "/etc/bind/db.255";
};
// add entries for other zones below here
};
```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 161

161

Les fichiers décrivant une zone

RFC 1035

```
#####$ cat /etc/bind/db.root
; This file holds the information on root name servers needed to
; initialize cache of Internet domain name servers
; This file is made available by InterNIC ...
; Sur ftp://ftp.rs.internic.net/domain/named.root.
;
3600000 IN NS A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET. 3600000 A 198.41.0.4
;
3600000 NS B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET. 3600000 A 128.9.0.107
;
; formerly C.PSI.NET
;
3600000 NS C.ROOT-SERVERS.NET.
C.ROOT-SERVERS.NET. 3600000 A 192.33.4.12
;
; .....
```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 162

162

Les fichiers décrivant une zone

```
#####$ cat /etc/bind/db.local
; BIND data file for local loopback interface
$TTL 604800
@ IN SOA localhost. root.localhost. (
    1 ; Serial
    604800 ; Refresh
    86400 ; Retry
    2419200 ; Expire
    604800 ) ; Negative Cache TTL
;
@ IN NS localhost.
@ IN A 127.0.0.1
#####$ cat /etc/bind/db.127
; BIND reverse data file for local loopback interface
$TTL 604800
@ IN SOA localhost. root.localhost. (
    1 ; Serial
    604800 ; Refresh
    86400 ; Retry
    2419200 ; Expire
    604800 ) ; Negative Cache TTL
;
@ IN NS localhost.
@ IN PTR 1.0.0.127.in-addr.arpa
```

@ désigne le nom de la zone (ici localhost)

@ désigne le nom de la zone (ici 127.in-addr.arpa)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 163

163

Partie 2 : Applications de l'Internet de type Client/Serveur (suite2)

Olivier GLÜCK
Université LYON 1/UFR d'Informatique
Olivier.Gluck@ens-lyon.fr
<http://www710.univ-lyon1.fr/~ogluck>



164

Plan de la partie 2

- Introduction / Rappel
- Connexions à distance (telnet/rlogin/rsh/ssh/X11)
- Applications de transfert de fichiers (FTP/TFTP)
- Accès aux fichiers distants (NFS/SMB)
- Gestion d'utilisateurs distants (NIS)
- DNS : un annuaire distribué
- **LDAP : un annuaire fédérateur sécurisé**
- La messagerie électronique (SMTP/POP/IMAP)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

165

165

LDAP : un annuaire fédérateur sécurisé



166

Problématique résolue par LDAP

- Permettre la fusion de multiples BD dans un unique annuaire informatique
 - base Microsoft Excel du personnel administratif
 - base Microsoft Access du personnel enseignant
 - base Microsoft Excel des numéros de téléphone
 - base `/etc/passwd` des comptes Unix des utilisateurs
 - base `/etc/aliases` (ou Sympa) de listes de Mail
 - base Samba des utilisateurs Windows
 - autres bases MySQL, Oracle, maps NIS,...
- Comment envoyer un mail à l'ensemble du personnel administratif sachant que l'administrateur système recevra uniquement une liste de (Nom, Prénom) ?

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

167

167

Le concept d'annuaire

- Annuaire informatique
 - service permettant d'accéder à des informations relatives à des personnes, des machines (ou autres ressources) de manière organisée
 - objectif : maintenir de façon cohérente et contrôlée une grande quantité de données
- Système de gestion de base de données (SGBD)
 - le schéma des données stockées est défini pour résoudre un certain problème ; il est connu des applis
 - les objets sont généralement complexes, stockés dans différentes tables ayant des relations entre elles
 - un langage spécifique permet la lecture et mise à jour des tables (requêtes SQL, ...)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

168

168

Le concept d'annuaire

- Différences annuaire/SGBD - dans un annuaire :
 - pas de liens de dépendances entre les objets stockés
 - les objets peuvent être distribués sur plusieurs annuaires pour assurer une meilleure disponibilité
 - le schéma de stockage des données est standardisé pour assurer un partage des données
 - les applications de l'annuaire n'ont pas besoin de connaître la structure interne des données stockées
 - un annuaire est principalement consulté en lecture et optimisé pour cela

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

169

169

L'annuaire LDAP

- LDAP : *Lightweight Directory Access Protocol*
- Héritier de l'annuaire X500 (proposée par l'ISO)
 - standard conçu par les opérateurs télécom pour interconnecter leurs annuaires téléphoniques
 - X500 adapté à Internet --> LDAP (même modèle de schéma, ...)
- Proposé à l'IETF en 1995
 - standard d'annuaire sur TCP/IP
 - le standard ne concerne pas le contrôle d'accès aux données de l'annuaire
 - Version 3 actuellement [RFC 2251]
 - Aussi : RFC 2252 à 2256, RFC 2829 à 2830, RFC 2849

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

170

170

L'annuaire LDAP

- Objectifs
 - fournir aux utilisateurs des informations fiables, facilement accessibles
 - permettre aux utilisateurs de mettre à jour eux-mêmes leurs informations personnelles
 - rendre les informations accessibles de façon contrôlée
 - faciliter le nomadisme des utilisateurs
 - éviter la redondance d'informations : un seul annuaire pour l'ensemble des services
 - faciliter la gestion (administration) des postes de travail, des équipements réseau
- sans remettre en cause les applications existantes !

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

171

171

L'annuaire LDAP

- Un modèle d'information : type des informations contenues dans l'annuaire
- Un protocole d'accès : comment accéder aux informations contenues dans l'annuaire
- Un modèle de nommage : comment l'information est organisée et référencée
- Un modèle fonctionnel : une syntaxe des requêtes permettant l'interrogation de la base et la mise à jour des informations
- Un modèle de duplication : comment la base est répartie sur différents serveurs (tolérance aux pannes, répartition de la charge du serveur, ...)
- Un modèle de sécurité : comment contrôler l'accès aux données ainsi que leur transfert

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

172

172

Le protocole LDAP

- Il définit
 - les échanges de la connexion Client/Serveur
 - commandes de connexion au service : `bind`, `unbind`, `abandon` (le client abandonne la requête en cours)
 - commandes de mises à jour des entrées de l'annuaire : `add`, `delete`, `modify`, `rename`
 - commandes d'interrogation : recherche (`search`) et comparaison (`compare`) d'entrées
 - le format de transport des données
 - pas de l'ASCII comme SMTP, HTTP, ...
 - encodage LBER : *Lightweight Basic Encoding Rules*

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

173

173

Le protocole LDAP

- Il définit
 - les échanges de la connexion Serveur/Serveur
 - la réplication (*replication service*), en cours de normalisation (*LDUP : LDAP Duplication Protocol*)
 - créer des liens entre différents annuaires (*referral service*) - défini dans LDAPv3
 - les mécanismes de sécurité
 - méthodes d'authentification pour se connecter à l'annuaire (qui peut se connecter à l'annuaire et comment)
 - mécanismes de règles d'accès aux données (une fois connecté, à quoi peut-on accéder et avec quels droits)
 - mécanismes de chiffrement des transactions

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

174

174

Le protocole LDAP

- LDAPv3 est conçu pour être extensible sans avoir à modifier la norme
 - permet l'ajout d'opérations (en plus des 9 de base)
 - permet l'ajout de paramètres associés à une opération
 - les mécanismes de sécurité sont définis dans une couche séparé : permet des méthodes d'authentification externes

```
ogluck@lima:/etc/ldap$ cat /etc/services | grep ldap
ldap 389/tcp # Lightweight Directory Access Protocol
ldap 389/udp # Lightweight Directory Access Protocol
ldaps 636/tcp # LDAP over SSL
ldaps 636/udp # LDAP over SSL
```

Olivier Glück

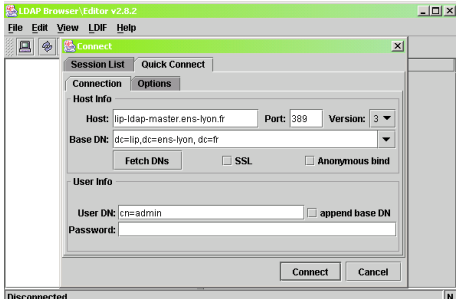
M2 SRS - Admin Systèmes et Réseaux

175

175

Se connecter à une base LDAP

Deux principaux éditeurs graphiques : GQ sous Unix (<http://biot.com/gq/>) et LDAP Browser/Editor sous Windows (<http://www.iit.edu/~gawojar/ldap/>)



176

176

Le modèle d'information

- Un annuaire est constitué de schémas LDAP qui vont déterminer les objets utilisables dans l'annuaire
- Un schéma LDAP
 - définit une liste des classes d'objets, les types des attributs et leur syntaxe répondant aux normes de l'*Object Management Group* (OMG)
 - standardisé (IANA) : pour l'interopérabilité entre logiciels
 - permet l'interfaçage avec les applications (Samba, ...)

`oqluck@lima:/etc/ldap$ ls /etc/ldap/schema/`
 README core.schema inetorgperson.schema krb5-kdc.schema
 nis.schema corba.schema cosine.schema java.schema
 misc.schema openldap.schema

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

177

177

Le modèle d'information

- Un attribut est défini par
 - un nom, un identifiant unique (OID), mono/multi-valué, une syntaxe et des règles de comparaison (*matching rules*), une valeur (format+taille limite), modifiable ou non
- Les classes d'objets modélisent
 - des objets réels : un compte Unix (`posixAccount`), une organisation (`o`), un département (`ou`), un personnel (`organizationPerson`), une imprimante (`device`), ...
 - ou abstraits : l'objet père de tous les autres (`top`), ...
- Une classe d'objet est définie par
 - un nom, un OID, des attributs obligatoires, des attributs optionnels, un type (structurel, auxiliaire ou abstrait)

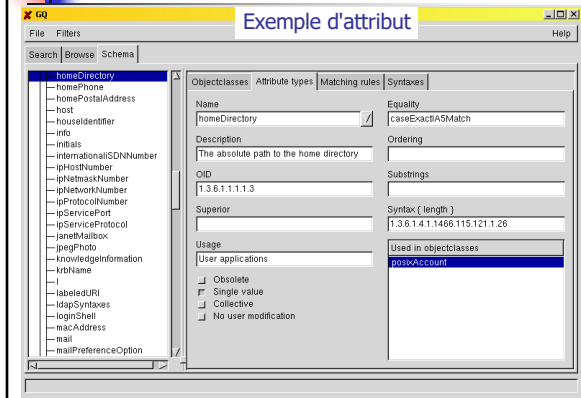
Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

178

178

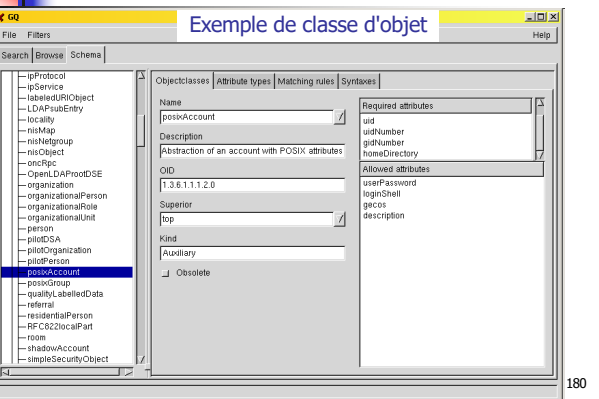
Le modèle d'information



179

179

Le modèle d'information

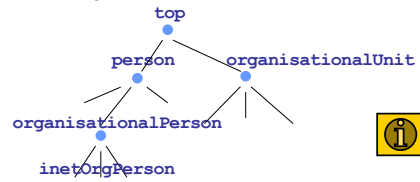


180

180

Le modèle d'information

- Les classes d'objets forment une structure arborescente : tout en haut, l'objet `top`



- Chaque objet hérite des attributs de l'objet dont il est le fils
- Plus d'infos :

<http://www.it.ufl.edu/projects/directory/ldap-schema/>
<http://ldap.akbkhome.com/>

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

181

181

Le modèle d'information

```

objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson

```

L'objet `person` a comme attributs : `commonName`, `surname`, `description`, `seeAlso`, `telephoneNumber`, `userPassword`

L'objet `file` `organizationalPerson` ajoute des attributs comme : `organizationUnitName`, `title`, `postalAddress`...

L'objet `petit-fils` `inetOrgPerson` lui rajoute des attributs comme : `mail`, `labeledURI`, `uid` (`userID`), `photo`...

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 182

182

Le modèle de nommage

- Il définit comment sont organisées les entrées (=objets) de l'annuaire et comment elles sont référencées
- Structure arborescente contenant deux catégories d'objets
 - les conteneurs (une zone de rangement) : départ d'une nouvelle branche
 - peuvent contenir des conteneurs ou des feuilles
 - généralement, une sous-organisation de l'organisation (département, zone géographique, ...)
 - les feuilles (véritables données) : terminaison des branches (généralement les machines, les utilisateurs, ...)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 183

183

Le modèle de nommage

- Structure logique hiérarchique : le DIT (*Directory Information Tree*)
- Une entrée est identifiée par un nom unique : le DN (*Distinguish Name*)
- RDN - *Relative Distinguish Name*

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 184

184

Le modèle de nommage

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 185

185

Le format LDIF

- LDAP Data Interchange Format* (LDIF)
- Standard de représentation des entrées sous format texte --> permet de
 - faire des imports/exports de la base ou d'une partie
 - créer, ajouter, modifier,... un grand nombre d'entrées de façon automatisée

```

dn: uid=ogluck,ou=People,dc=lip,dc=ens-lyon,dc=fr
objectClass: top
objectClass: account
objectClass: posixAccount
objectClass: shadowAccount
uid: ogluck
uidNumber: 44132
gidNumber: 200
homeDirectory: /home/toto
cn: Olivier GLUCK
loginShell: /bin/bash

```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 186

186

Le modèle fonctionnel

- Il décrit le moyen d'accéder aux données (syntaxe des requêtes) et les opérations que l'on peut leur appliquer
- Rappel des opérations de consultation/mise à jour
 - opérations de mise à jour des entrées de l'annuaire : `add`, `delete`, `modify`, `rename`
 - opérations d'interrogation : recherche (`search`) et comparaison (`compare`) d'entrées
 - > il n'y a pas d'opération de lecture d'une entrée : pour connaître le contenu d'une entrée, il faut écrire une requête qui pointe sur cette entrée

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 187

187

Le modèle fonctionnel

- Une requête est composée de 8 paramètres

base object	l'endroit de l'arbre où doit commencer la recherche
scope	la profondeur de la recherche
derefAliases	si on suit les liens ou pas
size limit	nombre de réponses limite
time limit	temps maxi alloué pour la recherche
attrOnly	renvoie ou pas la valeur des attributs en plus de leur type
search filter	le filtre de recherche
list of attributes	la liste des attributs que l'on souhaite connaître

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 188

188

Le modèle fonctionnel

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 189

189

Le modèle fonctionnel

- Les filtres de recherche [RFC 2254]

(<operator>(<search operation>)(<search operation>)...)

```
(mail=*) # existence
(uidNumber>=40000) # comparaison
(! (ou=People) (ou=Group)) # OU
(&(cn=Olivier*) (!(uid=ogluck))) # ET, contient, NON
(&(objectClass=posixGroup) (!(cn=reso*)(memberUid=*ogluck*)))
```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 190

190

Les URLs LDAP [RFC 1959]

- Permet aux clients Internet d'avoir un accès direct aux annuaires LDAP
- Syntaxe :

```
ldap[s]://<host>[:<port>]/<base_dn>?<attr>?<scope>?<filter>
```

<base_dn> : point de départ de la recherche
 <attr> : attributs consultés
 <scope> : étendue de la recherche (base, one, sub)
 <filter> : filtre de recherche (objectClass=*) par défaut

```
ldap://lip-ldap-master.ens-lyon.fr:389/dc=lip,dc=ens-lyon,dc=fr??sub?(cn=Olivier*)
```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 191

191

Les URLs LDAP [RFC 1959]

L'annuaire LDAP permet la mise à jour du carnet d'adresses

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 192

192

Le modèle de duplication

- Il définit comment dupliquer l'annuaire sur plusieurs serveurs
 - améliorer le temps de réponse
 - être tolérant aux pannes
- Deux types de serveurs LDAP
 - supplier serveur (maître) : fournit les données
 - consumer server (esclave) : reçoit les données du maître
- Possibilité de partitionner l'annuaire (éclatement sur plusieurs serveurs)
 - liens virtuels entre les différentes partitions (referral service)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 193

193

Le modèle de sécurité

- Authentification pour se connecter au service
 - Anonymous authentication, Root DN/passwd authentication (administrateur), User DN/passwd
- Contrôle de l'accès aux données
 - droits d'accès aux données (fonctions de l'utilisateur authentifié) : lecture d'une valeur (*read*), modification (*write*), recherche (*search*), comparaison (*compare*), ...
 - *search* : les données peuvent être une clé de recherche
 - *read* : permet de lire les données issues d'une recherche (par ex. *search* sur *cn* mais *read* seulement sur *Phone Number*)
 - règles définies sous forme d'ACLs (*Access Control List*) au niveau du sommet, d'un sous-arbre ou d'une entrée
- Chiffrement des transactions (LDAP+SSL, ...)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

194

194

Mettre en place un annuaire LDAP

- Il faut bien choisir les schémas
 - Quelles informations veut on stocker dans l'annuaire ?
 - --> choix des objets contenant les attributs désirés
 - Quelles sont les applications qui vont utiliser l'annuaire ?
 - Authentification des utilisateurs sous Unix, sous Windows (*samba*), gestion des groupes d'utilisateurs, listes de mail dynamiques (*sympa*), carnets d'adresses Netscape, ... ?
- Il faut réfléchir à l'organisation du DIT
 - impacts sur la performance, les droits d'accès, ...
- Puis dans un second temps
 - gestion centralisée sur un seul serveur ?
 - nombre de serveurs redondants ? Emplacement ?

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

195

195

OpenLDAP

- Logiciel LDAP du domaine public
- Le démon *slapd*
 - traite les requêtes LDAP
- Le démon *slurpd*
 - permet la réplication
- Des librairies LDAP
 - par exemple, pour authentifier les login via LDAP
libpam-ldap, *libnss-ldap*
- Des utilitaires
 - *ldapadd* *ldapdelete* *ldapmodify*
ldapmodrdn *ldappasswd* *ldapsearch*

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

196

196

Le fichier /etc/ldap/slapd.conf

- Permet de configurer le démon *slapd*
 - définition des schémas utilisés
`include /etc/ldap/schema/inetorgperson.schema`
 - définition du *backend* (moteur de base de données utilisé pour ranger les données)
`database ldbm (ldbm par défaut, sinon sql, ...)`
 - définition de la base de l'annuaire et de l'administrateur
 - le suffixe : racine de l'arbre
`suffix "dc=lip,dc=ens-lyon,dc=fr"`
 - l'administrateur et son mot de passe
`rootdn cn=admin,dc=lip,dc=ens-lyon,dc=fr`
`rootpw toto`
 - le répertoire où la base est stockée
`directory "/var/lib/ldap"`

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

197

197

Le fichier /etc/ldap/slapd.conf

- définitions des ACLs (man *slapd.access*)
 - Format d'un ACL :
`access to <what> [by <who> <access> [<control>]] +`
<what> : *, un dn, un filtre LDAP, une liste d'attributs (*attrs=...*)
<who> : *, dn, anonymous, users (quelqu'un authentifié), self (le proprio), ...
<access> : none, auth, compare, search, read, write, ...
<control> : stop, continue, break (imbrication des règles...)
■ Par défaut :
`access to attrs=userPassword`
 by dn="" write # l'admin
 by anonymous auth # droit de lecture uniquement lors du bind
 by self write # le propriétaire
 by * none
 - The admin dn has full write access
`access to *`
 by dn="" write
 by * read # nécessaire d'avoir read pour le bind

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

198

198

Le fichier /etc/ldap/slapd.conf

- définition des réplicats
 - sur le serveur maître
 - fichier dans lequel *slapd* stocke les modifications pour *slurpd*
`repllogfile /var/lib/ldap/repllog`
 - définition d'un réplicat
`replica host=ldap.ens-lyon.fr:389 bindmethod=...`
 - sur un esclave
 - le dn autorisé à faire la mise à jour
`updatedn "souvent slurpd"`
 - URL du maître
`updateref ldap://master-ldap.ens-lyon.fr:389`
- ... man *slapd.conf*

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

199

199

Le fichier /etc/ldap/ldap.conf

- Permet de donner des informations aux clients LDAP

- man ldap.conf
 - peut aussi être fait dans ~/.ldapprc
 - ou par des variables d'environnements
- ```
base par défaut à contacter pour les opérations LDAP
BASE dc=lip,dc=ens-lyon,dc=fr
en tant que qui le client se connecte à la base
BINDDN uid=ogluck,ou=People,dc=lip,dc=ens-lyon,dc=fr
le serveur auquel se connecter
HOST ldap.ens-lyon.fr:389
d'autres options de configuration...
```

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

200

200

## Ajouter des entrées dans l'annuaire

- Ecrire (ou générer) un fichier LDIF

```
lima/etc/ldap# cat structure.ldif
dn: dc=lima,dc=ens-lyon,dc=fr
objectClass: top
objectClass: organization
objectClass: dObject
description: Racine du DIT
ou: ENS/RESO
dc: lima

dn: ou=Members,dc=lima,dc=ens-lyon,dc=fr
objectClass: top
objectClass: organizationalUnit
description: Membres de RESO
ou: Members

dn: ou=Groups,dc=lima,dc=ens-lyon,dc=fr
objectClass: top
objectClass: organizationalUnit
description: Groupes dans RESO
ou: Groups

dn: ou=Hosts,dc=lima,dc=ens-lyon,dc=fr
objectClass: top
objectClass: organizationalUnit
description: Machines de RESO
ou: Hosts

lima/etc/ldap#

lima/etc/ldap# cat exemples.ldif
dn: uid=toto,ou=Members,dc=lima,dc=ens-lyon,dc=fr
objectClass: top
objectClass: account
objectClass: posixAccount
objectClass: shadowAccount
uid: toto
uidNumber: 44132
gidNumber: 200
homeDirectory: /home/toto
cn: TOTO
loginShell: /bin/bash

dn: cn=tata,ens-lyon.fr,ou=Hosts,dc=lima,dc=ens-lyon,dc=fr
objectClass: top
objectClass: device
objectClass: ipHost
ipHostNumber: 132.227.71.30
cn: tata,ens-lyon.fr
cn: tata

dn: cn=admin_group,ou=Groups,dc=lima,dc=ens-lyon,dc=fr
objectClass: top
objectClass: organizationalUnit
description: Machines de RESO
memberUid: toto
cn: admin_group
lima/etc/ldap#
```

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

201

201

## Ajouter des entrées dans l'annuaire

- Utiliser la commande ldapadd

```
lima/etc/ldap#
lima/etc/ldap# ldapadd -x -h localhost -p 389 -D "cn=admin,dc=lima,dc=ens-lyon,dc=fr" -W -f structure.ldif
Enter LDAP Passwords
ldap_bind: Invalid credentials (49)
lima/etc/ldap# ldapadd -x -h localhost -p 389 -D "cn=admin,dc=lima,dc=ens-lyon,dc=fr" -W -f structure.ldif
Enter LDAP Passwords
adding new entry "dc=lima,dc=ens-lyon,dc=fr"

adding new entry "ou=Members,dc=lima,dc=ens-lyon,dc=fr"

adding new entry "ou=Groups,dc=lima,dc=ens-lyon,dc=fr"

adding new entry "ou=Hosts,dc=lima,dc=ens-lyon,dc=fr"

lima/etc/ldap# ldapadd -x -h localhost -p 389 -D "cn=admin,dc=lima,dc=ens-lyon,dc=fr" -W -f exemples.ldif
Enter LDAP Passwords
adding new entry "uid=toto,ou=Members,dc=lima,dc=ens-lyon,dc=fr"
ldap_add: Already exists (68)

ldif_record() = 68
lima/etc/ldap# ldapadd -x -h localhost -p 389 -D "cn=admin,dc=lima,dc=ens-lyon,dc=fr" -W -f exemples.ldif
Enter LDAP Passwords
adding new entry "uid=toto,ou=Members,dc=lima,dc=ens-lyon,dc=fr"

adding new entry "cn=tata,ens-lyon.fr,ou=Hosts,dc=lima,dc=ens-lyon,dc=fr"

adding new entry "cn=admin_group,ou=Groups,dc=lima,dc=ens-lyon,dc=fr"

lima/etc/ldap#
```

202

202

## Ajouter des entrées dans l'annuaire

- Utiliser un client LDAP permettant l'ajout d'entrées

203

203

## Interroger l'annuaire ldapsearch

```
xterm
toto@lima:~$ ldapsearch -x -h localhost -p 389 -D "uid=toto,ou=Members,dc=lima,dc=ens-lyon,dc=fr" -W
-s sub -b "ou=Members,dc=lima,dc=ens-lyon,dc=fr"
Enter LDAP Passwords
version: 2

filter: (objectclass=*)
requesting: ALL

Members, lima, ens-lyon, fr
dn: ou=Members,dc=lima,dc=ens-lyon,dc=fr
objectClass: top
objectClass: organizationalUnit
description: Membres de RESO
ou: Members

toto, Members, lima, ens-lyon, fr
dn: uid=toto,ou=Members,dc=lima,dc=ens-lyon,dc=fr
objectClass: top
objectClass: account
objectClass: posixAccount
objectClass: shadowAccount
uid: toto
uidNumber: 44132
gidNumber: 200
homeDirectory: /home/toto
loginShell: /bin/bash
userPassword: dgFW0z=
cn: TOTO

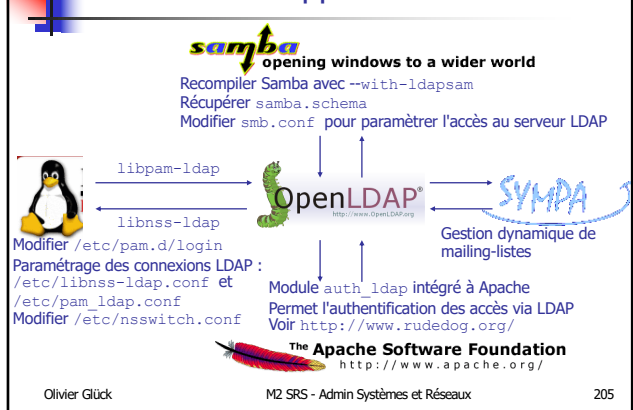
search result
search: 2
result: 0 Success

numResponses: 3
numEntries: 2
toto@lima:~$
```

204

204

## Liens avec les applications



Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

205

205

## Authentification Unix via LDAP

- PAM - *Pluggable Authentication Modules*
  - permet de gérer la politique d'authentification des connexions sans recompiler quoi que ce soit
  - pour authentification via LDAP, rajouter la ligne `auth sufficient pam_ldap.so` dans le fichier `/etc/pam.d/login` qui signifie l'authentification via LDAP est suffisante
  - voir aussi `/etc/pam.d/ssh`, `/etc/pam.d/rsh` ...
- Configurer l'accès à la base LDAP dans
  - `/etc/libnss-ldap.conf` et `/etc/pam_ldap.conf`
  - voir les pages man associées
- Indiquer dans `/etc/nsswitch.conf` l'ordre d'interrogation pour l'authentification
  - toujours laisser `files` en premier !

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 206

206

## Authentification Unix via LDAP

The image shows two terminal windows. The left window shows the output of `cat /etc/ldap/ldap.conf` and `ping tata`. The right window shows the output of `cat /etc/nsswitch.conf`.

```

lima/etc/ldap
lima/etc/ldap#id toto
uid=44132(toto) gid=200(lip) groups=200(lip),259(admin_group)
lima/etc/ldap#telnet localhost
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
Debian GNU/Linux 3.0 with bunk-1 packages by Adrian Bunk <bunk@
lima login: toto
Password:
Last login: Sat Apr 3 23:18:26 2004 from localhost on pts/5
Linux lima 2.4.22-1-686 #6 Sat Oct 4 14:09:08 EST 2003 i686 GNU
toto@lima:~$ ls -ld /home/toto
drwxr-xr-x 2 toto lip 1024 Apr 3 23:07 /home/toto
toto@lima:~$ chgrp admin_group /home/toto
toto@lima:~$ ls -ld /home/toto
drwxr-xr-x 2 toto admin_group 1024 Apr 3 23:07 /home/toto
toto@lima:~$ ping tata
PING tata.ens-lyon.fr (132.227.71.30): 56 data bytes

--- tata.ens-lyon.fr ping statistics ---
1 packets transmitted, 0 packets received, 100% packet loss
toto@lima:~$ ping tata.ens-lyon.fr
PING tata.ens-lyon.fr (132.227.71.30): 56 data bytes

--- tata.ens-lyon.fr ping statistics ---
1 packets transmitted, 0 packets received, 100% packet loss
toto@lima:~$

```

```

lima/etc/ldap#cat /etc/nsswitch.conf
/etc/nsswitch.conf
#
Example configuration of GNU Name S
Information about this file is avai
passwd: files ldap
group: files ldap
shadow: files ldap
hosts: files dns ldap
toto@lima:~$

```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 207

207

## Authentification Unix via LDAP

The image shows a terminal window where the user performs an LDAP search and updates their password.

```

toto@lima:~$ head -4 /etc/pam.d/passwd
#
The PAM configuration file for the Shadow 'passwd' service
#
passwd sufficient pam_ldap.so
toto@lima:~$ ldsearch -x -h localhost -p 389 -D "uid=toto,ou=Members,dc=lima,dc=ens-lyon,dc=fr" -W -LLL
-s sub -b "uid=toto,ou=Members,dc=lima,dc=ens-lyon,dc=fr" "(cn=TOTO*)" userPassword
Enter LDAP Password:
dn: uid=toto,ou=Members,dc=lima,dc=ens-lyon,dc=fr
userPassword:: d030bu==

toto@lima:~$ passwd
Enter login(LDAP) password:
New password:
Re-enter new password:
LDAP password information changed for toto
passwd: password updated successfully
toto@lima:~$ ldsearch -x -h localhost -p 389 -D "uid=toto,ou=Members,dc=lima,dc=ens-lyon,dc=fr" -W -LLL
-s sub -b "uid=toto,ou=Members,dc=lima,dc=ens-lyon,dc=fr" "(cn=TOTO*)" userPassword
Enter LDAP Password:
dn: uid=toto,ou=Members,dc=lima,dc=ens-lyon,dc=fr
userPassword:: e1NNRDj9TKfZcFR1MD1zVlpuOHVjNFBoNzZubHB1a09RPQ==
toto@lima:~$

```

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 208

208

## Authentification Samba via LDAP

- Dans `/etc/samba/smb.conf`

```

[global]
paramétrage des connexions LDAP
ldap server = localhost
ldap port = 389
ldap suffix = "dc=lima,dc=ens-lyon,dc=fr"
ldap admin dn = "cn=admin,dc=lima,dc=ens-lyon,dc=fr"
ldap ssl = no

```
- Après avoir créé une entrée `sambaAccount` dans l'annuaire pour `user_login`, il suffit de faire `smbpasswd user_login` pour que Samba mette à jour les champs Samba dans l'annuaire

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 209

209

## Partie 2 : Applications de l'Internet de type Client/Serveur (suite3)

Olivier GLÜCK  
 Université LYON 1/UFR d'Informatique  
 Olivier.Gluck@ens-lyon.fr  
 http://www710.univ-lyon1.fr/~olgluck

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 210

210

## Plan de la partie 2


- Introduction / Rappel
- Connexions à distance (telnet/rlogin/rsh/ssh/X11)
- Applications de transfert de fichiers (FTP/TFTP)
- Accès aux fichiers distants (NFS/SMB)
- Gestion d'utilisateurs distants (NIS)
- DNS : un annuaire distribué
- LDAP : un annuaire fédérateur sécurisé
- **La messagerie électronique (SMTP/POP/IMAP)**
- Le protocole HTTP

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 211

211

## La messagerie électronique

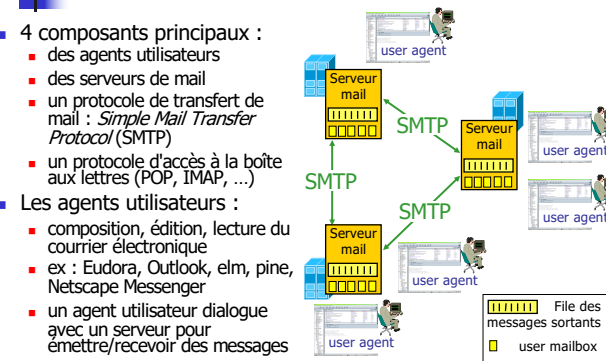
Les différents composants  
 Configuration d'un agent utilisateur  
 Le protocole SMTP  
 Codage des messages et types MIME  
 Les protocoles d'accès



212

## Courrier électronique : les composants

- 4 composants principaux :
  - des agents utilisateurs
  - des serveurs de mail
  - un protocole de transfert de mail : *Simple Mail Transfer Protocol* (SMTP)
  - un protocole d'accès à la boîte aux lettres (POP, IMAP, ...)
- Les agents utilisateurs :
  - composition, édition, lecture du courrier électronique
  - ex : Eudora, Outlook, elm, pine, Netscape Messenger
  - un agent utilisateur dialogue avec un serveur pour émettre/recevoir des messages

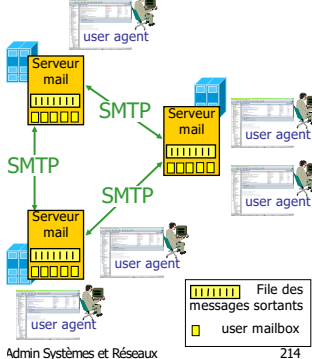


Olivier Glück M2 SRS - Admin Systèmes et Réseaux 213

213

## Courrier électronique : les composants

- Les messages entrants et sortants sont stockés sur le serveur
- La boîte aux lettres de chaque utilisateur contient les messages entrants (à lire)
- File d'attente de messages mail sortants (à envoyer)
- Protocole SMTP entre les serveurs de mail pour l'envoi des messages
  - modèle C/S : Client (serveur de mail émetteur) - Serveur (serveur de mail récepteur)
  - le client se connecte sur le port 25/TCP du serveur pour transférer son message

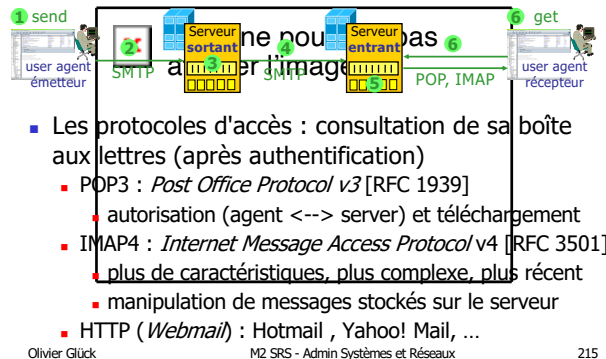


Admin Systèmes et Réseaux 214

214

## Courrier électronique : les composants

- Les protocoles d'accès : consultation de sa boîte aux lettres (après authentification)
  - POP3 : *Post Office Protocol v3* [RFC 1939]
    - autorisation (agent <--> server) et téléchargement
  - IMAP4 : *Internet Message Access Protocol v4* [RFC 3501]
    - plus de caractéristiques, plus complexe, plus récent
    - manipulation de messages stockés sur le serveur
  - HTTP (*Webmail*) : Hotmail, Yahoo! Mail, ...




Olivier Glück M2 SRS - Admin Systèmes et Réseaux 215

215

## Courrier électronique : les composants

- Dans les débuts du courrier électronique
  - il n'y avait pas de protocole d'accès
  - SMTP était juste prévu pour échanger du courrier entre le serveur de l'émetteur (client) et le serveur du récepteur (serveur) --> étapes 3-4-5 uniquement
- Pourquoi un protocole d'accès et une évolution de SMTP permettant au serveur sortant d'être à la fois client et serveur SMTP ?
- Pourquoi une file des messages sortants ?
- Pourquoi ne pas mettre le serveur sortant directement sur le poste utilisateur ?

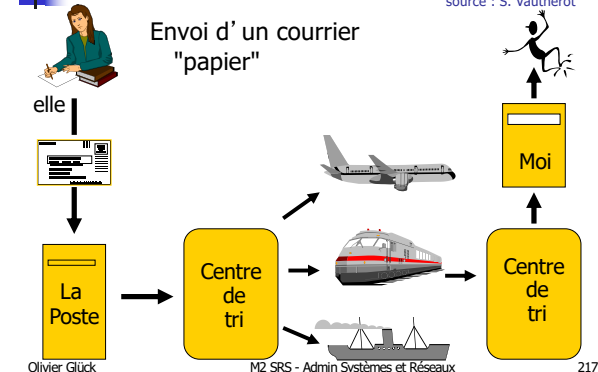


216

216

## Analogie : le courrier "papier"

source : S. Vautherot



Olivier Glück M2 SRS - Admin Systèmes et Réseaux 217

217

## Configuration d'un agent utilisateur

L'identité permet de renseigner une partie de l'en-tête des messages envoyés

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 218

218

## Configuration d'un agent utilisateur

Paramétrage du serveur sortant

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 219

219

## Configuration d'un agent utilisateur

POP : les messages sont rapatriés dans le répertoire local

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 220

220

## Configuration d'un agent utilisateur

IMAP : les messages restent sur le serveur sauf s'ils sont supprimés, déplacés, ...

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 221

221

## Le protocole SMTP [RFC 821]

- Transfert direct entre le serveur émetteur et le serveur récepteur (port 25/TCP)
- 3 phases de transfert
  - handshaking (établissement de la connexion)
  - transfert d'un ou plusieurs messages
  - fermeture de la connexion
- Les connexions sont **persistentes**
  - si plusieurs messages à destination du même serveur sont en attente sur le serveur émetteur, ils transiteront tous sur la même connexion TCP

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 222

222

## Le protocole SMTP [RFC 821]

- Un message est composé d'un en-tête et d'un corps (RFC 822)
  - les champs de l'en-tête peuvent être positionnés soit par l'agent utilisateur émetteur, soit par le serveur entrant, soit par le serveur sortant
  - un champ d'en-tête est de la forme  
nom\_champ: valeur<CRLF>
  - l'en-tête contient au minimum les champs From et To, très souvent le champ Subject
  - peut permettre de mettre en place des filtres...

plus d'infos : <http://www.cru.fr/messagerie/accents.html>

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 223

223

## Le protocole SMTP [RFC 821]

- Une succession de Commande/Réponse
  - Commande SMTP : texte ASCII
  - Réponse SMTP : code d'état (status) + phrase
- Un message peut contenir plusieurs objets ; ils sont alors envoyés dans un message "multipart" (contrairement à HTTP : 1 objet = 1 réponse)
- Le serveur SMTP utilise `CRLF.CRLF` pour reconnaître la fin d'un message
- Les messages (en-tête ET corps) sont transférés en ASCII 7 bits (US-ASCII)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

224

224

## Le codage des messages

- SMTP est prévu pour transférer des caractères US-ASCII sur 7 bits --> problème de la représentation des caractères accentués, du transfert des octets (images...)
- Pour transférer une image ou du texte accentué, l'agent utilisateur émetteur/récepteur doit encoder/décoder le contenu du message
- Encodage `quoted-printable` :
  - généralement utilisé pour transférer du texte
  - permet le transfert des caractères ASCII étendus (codés sur 8 bits >128) comme les caractères accentués :
    - ils sont codés par les 3 caractères US-ASCII suivants : `=xx` où `xx` est le code hexadécimal du caractère à encoder
    - du coup, il faut coder le caractère `=` différemment : `=20`

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

225

225

## Le codage des messages

- Encodage `base64` :
  - généralement utilisé pour transférer des flux d'octets
  - permet le transfert des images ou autre série d'octets en tant que caractères ASCII NVT :
    - 3 octets (24 bits) sont transférés en tant que 4 caractères ASCII NVT : les 3 octets sont découpés en 4 fois 6 bits
    - bourrage avec le caractère `=` si pas aligné sur 4 caractères
    - permet de ne pas transférer plus de bits que le contenu initial (excepté le bourrage)
- ESMTP [RFC 1425] : une évolution de SMTP qui permet le transfert des messages sans passer au format ASCII NVT
  - transfert de blocs de données sur 8 bits (flux d'octets)
  - spécifié par `Content-Transfer-Encoding: 8bit OU Binary` dans l'en-tête

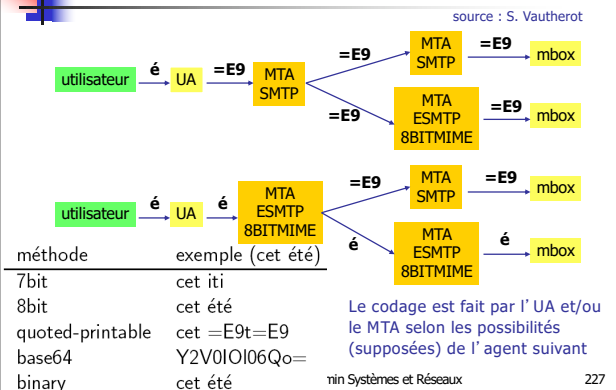
Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

226

226

## Le codage des messages



min Systèmes et Réseaux

227

227

## Les types MIME [RFC 2045, 2056]

- MIME : *Multi-purpose Internet Mail Extensions*
- Permet l'échange de fichiers multimédias entre machines quelconques en spécifiant dans l'en-tête
  - le type du fichier en vue d'un traitement par l'agent utilisateur destinataire
  - le codage des données du fichier
- Les commandes MIME ont été intégrées dans HTTP1.0
- Un type MIME est composé
  - d'un type général (text, image, audio, video, application...)
  - et d'un sous-type (image/gif, image/jpeg, application/pdf, application/rtf, application/msword, text/plain, text/html)
- En perpétuelle évolution
- La machine cliente doit ensuite associer l'exécution d'une application à chaque type MIME

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

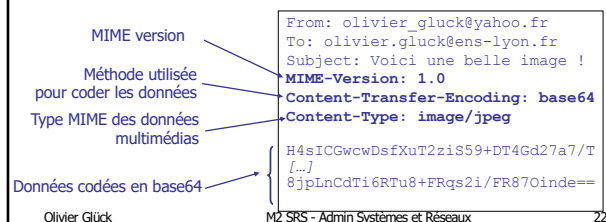
228

228

## Les types MIME [RFC 2045, 2056]

### Content-Type: type/subtype; parameters

- Lignes supplémentaires dans l'en-tête du message pour déclarer un type MIME et un encodage
- Content-type est généralement positionné à partir de l'extension du document demandé (`/etc/mime.types`)



Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

229

229

## Le type Multipart

```
From: olivier_gluck@yahoo.fr
To: olivier_gluck@ens-lyon.fr
Subject: Voici une belle image mais avec du texte !
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary=98766789

--98766789
Content-Transfer-Encoding: quoted-printable
Content-Type: text/plain

Cher Olivier,
Voici une photo de nos dernieres vacances !
--98766789
Content-Transfer-Encoding: base64
Content-Type: image/jpeg

H4sICGYRMTQAA3NsaWRlcy5wcvDsfXuT2ziS59+DT4Gd275a
56o7LlgsJbFNiWpSqsfw6rvLxPgSxILVnk64i54ftrKI67/T
[...]
8jplnCdTi6RTu8+FRqs2i/RTuy56plYbVvsalfdvUjHrtV6g
RTf4/hy67fgIIVDFer+rtYuNFR870inde==
--98766789--
```

Olivier Gluck

M2 SRS - Admin Systèmes et Réseaux

230

230

## Les commandes SMTP

| Commande                 | Description                                     |
|--------------------------|-------------------------------------------------|
| <b>HELO</b> nom_client   | identifie le client SMTP ; établit la connexion |
| <b>MAIL</b> From: <@exp> | identifie l'expéditeur du message               |
| <b>RCPT</b> To: <@dest>  | désigne le destinataire du message              |
| <b>DATA</b>              | indique le début du message (en-tête+corps)     |
| <b>QUIT</b>              | termine la connexion                            |
| <b>NOOP</b>              | pas d'opération ; force le serveur à répondre   |
| <b>RSET</b>              | réinitialisation de la saisie de données (DATA) |

```
xterm
ogluck@lima:~$ telnet localhost 25
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
220 lima.cri2000.ens-lyon.fr ESMTP Exin 3.35 #1 Mon, 22 Mar 2004 11:57:58 +0100
HELP
214-Commands supported:
214- HELO EHLO MAIL RCPT DATA AUTH
214 NOOP QUIT RSET HELP
QUIT
221 lima.cri2000.ens-lyon.fr closing connection
Connection closed by foreign host.
ogluck@lima:~$
```

231

231

## Un échange SMTP

"Nice to meet you !"

... sender OK

... receiver OK

Début de l'en-tête (DATA)

Fin de l'en-tête (ligne vierge)

Fin du message (<CR><LF> . <CR><LF>)

```
xterm
ogluck@lima:~$ telnet mailhost.ens-lyon.fr 25
Trying 140.77.11.22...
Connected to oceanite.ens-lyon.fr.
Escape character is '^]'.
HELO lima.ens-lyon.fr
250 oceanite.ens-lyon.fr ESMTP Postfix
MAIL FROM: ogluck
250 Ok
RCPT TO: <olivier_gluck@ens-lyon.fr>
250 Ok
RCPT TO: <ogluck@bat710.univ-lyon1.fr>
250 Ok
DATA
354 End data with <CR><LF> . <CR><LF>
From: olivier_gluck@yahoo.fr
To: olivier_gluck@bat710.univ-lyon1.fr
Cc: GLUCK <olivier_gluck@ens-lyon.fr>
Subject: un dialogue SMTP

Voici un exemple d'echange !
250 Ok: queued as 119F332015D
MAIL FROM: ogluck
250 Ok
RCPT TO: <olivier_gluck@yahoo.fr>
250 Ok
DATA
354 End data with <CR><LF> . <CR><LF>
From: olivier_gluck@ens-lyon.fr
To: olivier_gluck@yahoo.fr

Un deuxième mail à envoyer...
250 Ok: queued as 1A1863200F6
QUIT
221 Bye
Connection closed by foreign host.
ogluck@lima:~$
```

232

## Exemple d'en-tête

| Sujet                                                   | Expéditeur               | Date             | Etat     |
|---------------------------------------------------------|--------------------------|------------------|----------|
| Cours SMTP                                              | Olivier GLUCK            | 19:17            | Lire     |
| Réunion d'information : les journées MAGE sort à Lyon ! | planification@jrm2004... | 18/03/2004 13... | Lire     |
| Home page                                               | admin@bat710.univ-ly...  | 18/03/2004 12... | Répondre |

**Sujet: Cours SMTP**

**De:** Olivier GLUCK <Olivier\_Gluck@ens-lyon.fr>

**Date:** 19:17

**A:** Olivier GLUCK <ogluck@bat710.univ-lyon1.fr>, Olivier GLUCK <Olivier\_Gluck@ens-lyon.fr>

**X-UID:** eb9d1caa7080000

**X-Mozilla-Status:** 0001

**X-Mozilla-Status2:** 00000000

**Received:** from ens-lyon.fr (gluck.cri2000.ens-lyon.fr [140.77.13.102]) by oceanite.ens-lyon.fr (Postfix) with SMTP id 3CC61320118; Fri, 19 Mar 2004 19:17:26 +0100 (CET)

**X-From:** Olivier\_Gluck@ens-lyon.fr Fri Mar 19 19:17:38 2004

**Return-Path:** <olivier\_gluck@ens-lyon.fr>

**Message-ID:** <40583945.2050607@ens-lyon.fr>

**Organization:** Université LYON 1/Equipe RESO (LIP ENS Lyon)

**User-Agent:** Mozilla/5.0 (Windows; U; Windows NT 5.1; fr-FR; rv:1.0.2) Gecko/20030208 Netscape/7.0.2

**X-Accept-Language:** fr-fr, fr

**MIME-Version:** 1.0

**Content-Type:** text/plain; charset=ISO-8859-15; format=flowed

**Content-Transfer-Encoding:** 8bit

**X-Virus-Scanned:** by AMaVSi snapshot-20020222

Voilà un exemple !

233

## Exemple de contenu d'une bal

```
SunOS:/users/cao/glucko
glucko@ducas [SunOS] ~$ cat /var/mail/ogluck
From: Olivier_Gluck@numericable.fr Mon Mar 22 20:04:51 2004
Return-Path: <Olivier_Gluck@numericable.fr>
Received: from isis.lip6.fr (isis.lip6.fr [132.227.60.21])
by asin.lip6.fr (8.11.6p3/8.11.6) with ESMTP id i2M34pM14034
for <Olivier_Gluck@asin.lip6.fr>; Mon, 22 Mar 2004 20:04:51 +0100 (CET)
Received: from oglightred.numericable.net (oglightred.numericable.net [190.236.0.153])
by isis.lip6.fr (8.12.11/tpda-5.4+vector) with ESMTP id i2M34pI3032651
for <Olivier_Gluck@lip6.fr>; Mon, 22 Mar 2004 20:04:51 +0100
X-pt: isis.lip6.fr
Received: (qmail 15060 invoked from network); 22 Mar 2004 19:04:45 -0000
Received: from unknown (HELO numericable.fr) ([81.220.146.234])
(envelope-sender <Olivier_Gluck@numericable.fr>)
by oglightred.numericable.net (qmail-1dgr-1.03) with SMTP
for <Olivier_Gluck@lip6.fr>; 22 Mar 2004 19:04:45 -0000
Message-ID: <405F3886.7020804@numericable.fr>
Date: Mon, 22 Mar 2004 20:04:54 +0100
From: Olivier GLUCK <Olivier_Gluck@numericable.fr>
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; fr-FR; rv:1.0.2) Gecko/20030208 Netscape/7.0.2
X-Accept-Language: fr-fr, fr
MIME-Version: 1.0
To: Olivier Gluck <Olivier_Gluck@lip6.fr>
Subject: Cours SMTP
Content-Type: text/plain; charset=ISO-8859-15; format=flowed
Content-Transfer-Encoding: 8bit
X-Scanned-By: isis.lip6.fr

Voilà le contenu d'une BAL contenant 1 seul message !

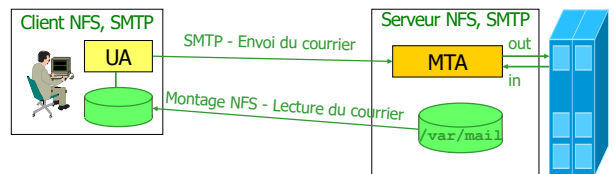
glucko@ducas [SunOS] ~$
```

Une BAL n'est rien de plus qu'un fichier ! (généralement /var/mail/user\_login)

234

## L'accès à sa boîte aux lettres

- Par lecture directe du fichier (en local ou par montage NFS)



Olivier Gluck

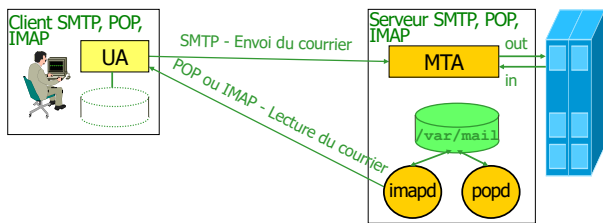
M2 SRS - Admin Systèmes et Réseaux

235

235

## L'accès à sa boîte aux lettres

- En utilisant un protocole spécifique (POP, IMAP) ou le protocole HTTP qui traverse généralement les pare-feus



Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

236

236

## Le protocole POP3 [RFC 1939]

### Phase d'autorisation

#### Commandes client :

- user**: déclare username
- pass**: password

#### Deux réponses possible du serveur :

- +OK
- ERR

### Phase de transaction

- list**: liste les numéros de messages et leur taille
- retr**: rappatrie un message à partir de son numéro
- dele**: efface un message
- quit**

```
S: +OK POP3 server ready
C: user alice
S: +OK
C: pass hungry
S: +OK user successfully logged on

C: list
S: 1 498
S: 2 912
S: .
C: retr 1
S: <contenu du message 1>
S: .
C: dele 1
C: retr 2
S: <contenu du message 2>
S: .
C: dele 2
C: quit
S: +OK POP3 server signing off
```

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

237

237

## Le protocole POP3 [RFC 1939]

- POP3 est extrêmement simple
  - permet uniquement de télécharger des messages depuis le serveur en laissant éventuellement une copie de ceux-ci dans la BAL de l'utilisateur
  - pas adapté aux utilisateurs nomades
    - impossible de gérer des répertoires sur le serveur
    - impossible de gérer les messages en les laissant sur le serveur

--> IMAP répond à cette problématique au prix d'un protocole beaucoup plus complexe

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

238

238

## Le protocole IMAP [RFC 3501]

- IMAP permet la gestion distante des messages
  - associe un message à un répertoire distant sur le serveur
  - permet à l'utilisateur de faire une recherche dans les messages sur le serveur
  - permet de ne consulter que des extraits de messages (par exemple que l'en-tête ou que la partie texte d'un message *multipart*...)
  - contrairement à POP3, IMAP conserve des informations d'état sur chaque utilisateur (noms des répertoires, listes des messages qu'ils contiennent...)

Plus d'infos : <http://www.imap.org/>

<http://cri.univ-lyon2.fr/doc/ImapMaisCEstTresSimple.html>

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

239

239

## L'accès Webmail

- Pas de protocole d'accès spécifique
  - l'utilisateur utilise un navigateur Web comme agent utilisateur pour consulter/envoyer ses courriers
- Utilise le protocole HTTP (ou HTTPS) pour communiquer avec les serveurs SMTP/IMAP
  - le serveur HTTP exécute des scripts qui utilisent
    - le protocole IMAP pour communiquer avec le serveur IMAP et ainsi manipuler les messages distants de l'utilisateur
    - le protocole SMTP pour traduire une demande d'envoi d'un message de la part de l'utilisateur

### Avantages

- adapté aux utilisateurs itinérants
- pas besoin d'un agent utilisateur particulier, seule une connexion Internet avec Navigateur Web est nécessaire

Plus d'infos : <http://www.cru.fr/http-mail/critere.html>

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

240

240

## Les alias

- Adresse d'un destinataire : bal@nom\_domaine
- Problème :
  - bal n'est pas forcément le login de l'utilisateur
  - nom\_domaine n'est pas forcément le nom du serveur de mail contenant les BAL
  - bal peut représenter plusieurs destinataires (listes)
- Il faut faire des alias (souvent /etc/aliases)
 

```
Olivier.Gluck --> /var/mail/ogluck
ens-lyon.fr --> mailhost.ens-lyon.fr
```

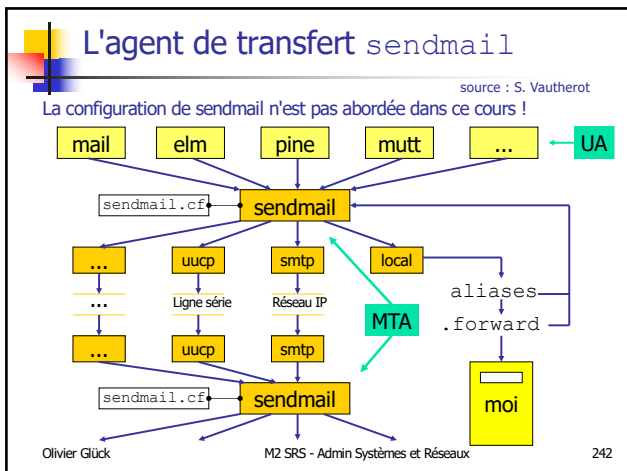
Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

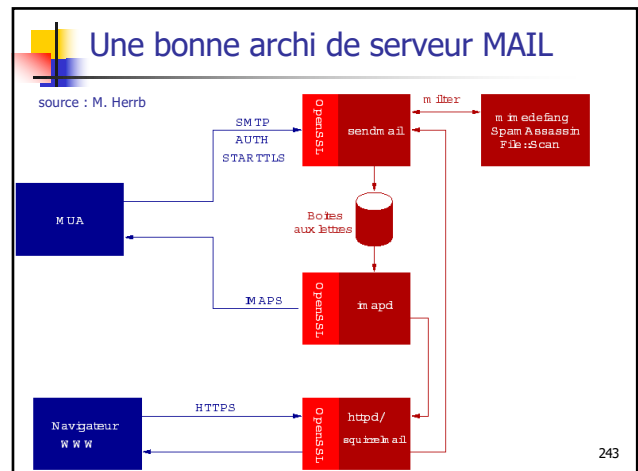
241

241





242



243

## HTTP : le protocole du Web

Intro : Web, URL et Formulaires  
Format des requêtes/réponses  
Durée de vie des connexions, Cookies  
Différentes versions de HTTP, Proxy  
Les requêtes clientes, les réponses du serveur  
Les en-têtes, les types MIME  
CGI, GET/POST, Format URL-encodé

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 244

244

## Les services d'Internet

- Un service = une **application** qui utilise un **protocole** et un numéro de **port**
- Fonctionnement en mode **Client/Serveur** au dessus de TCP/IP

**Client** (E-mail, telnet, ftp, news, http, ...)

**Le Web** (Navigateur, Windows, Modem ADSL)

**Le Web** (Serveur Apache, Linux, Ethernet)

**Internet** (Types de services différents, Un protocole par service)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 245

245

## World Wide Web

- Architecture pour accéder à des documents liés entre eux et situés sur des machines reliées par Internet
- Architecture basée sur 3 concepts :
  - la localisation --> **URL**
  - le protocole --> **HTTP**
  - le langage --> **HTML**
- Popularité due à :
  - interfaces graphiques conviviales
  - très grande quantité d'informations
  - grande diversité des informations

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 246

246

## Le jargon du Web

- Une page Web :
  - contient des "objets"
  - désignée par une adresse (URL)
- La plupart des pages Web contiennent :
  - du code HTML de base
  - des objets référencés
- L'URL a au moins deux composantes :
  - le nom d'hôte contenant la page Web
  - le chemin d'accès sur l'hôte
- L'Agent Utilisateur pour le Web est le **browser** :
  - MS Internet Explorer
  - Netscape Communicator
  - ...
- Le serveur Web :
  - Apache (domaine public)
  - MS Internet Information Server

[www.someSchool.edu/someDept/pic.gif](http://www.someSchool.edu/someDept/pic.gif)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 247

247

## Origines du Web

- Naissance au CERN : besoin d'échanges de documents, rapports, croquis, photos... entre des grosses équipes internationales pour des expériences demandant de longs investissements de mise en œuvre
  - mars 89 : Tim Berners-Lee : réseau de documents
  - septembre 90 : 1er prototype (mode texte)
  - décembre 91 : démonstration publique à la conférence Hypertext'91 de San Antonio

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

248

248

## Envol du Web

- Février 93 : 1ère interface graphique *Mosaic* (Marc Andreessen)
- 1994 : M. Andreessen crée *Netscape Comm. Corp.* (développements logiciels pour le web)
- 1994 : création du W3C (**WWW Consortium**) par le CERN et le MIT (Tim Berners-Lee président) (développements du Web, standards...)
- 1996 : apparition des feuilles de styles (CSS)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

249

249

## Fonctionnement du Web

- Le client (navigateur ou *browser*) dialogue avec un serveur Web selon le protocole HTTP
- Le serveur vérifie la demande, les autorisations et transmet l'information
- Le navigateur interprète le fichier reçu et l'affiche (le navigateur, un *plug-in* ou un *helper*)
- A ce schéma de base, peuvent s'ajouter :
  - des **contrôles** par compte individuel, par domaine, par adresse IP...
  - des **exécutions** de code coté serveur et/ou coté client

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

250

250

## Adressage des documents

- Il faut nommer, localiser et accéder à une page :  
--> 3 questions : Quoi ? Où ? Comment ?
- Solution :
  - URL - *Uniform Resource Locator* : Adresse universelle de ressource
    - en 3 parties : le protocole (comment), le nom DNS (où) et le nom du document (quoi)
  - URL --> URI (*Universal Resource Identifier*)
    - un sur-ensemble des URLs
  - URL classique (simplifiée) :

<http://www.monsite.fr/projet/doc.html>

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

251

251

## Adressage des documents

- Différentes composantes d'une URL :  
`proto://host_name:port/path/extra_path?arguments`
  - la racine "/" de `path` est définie par la configuration du serveur Web  
(**Attention** : à ne pas confondre avec la racine du système de fichiers sur le serveur)
  - `/path` peut contenir une étiquette (point d'ancrage)  
`http://www.monsite.fr/projet/doc.html#label`
  - `extra_path` (après `.cgi` par ex.) et `arguments` permettent de passer des informations à des programmes s'exécutant sur le serveur

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

252

252

## Adressage des documents

- URL relative :
  - un lien vers "images/new.gif" dans la page  
`http://www.monsite.fr/projet/doc.html`  
est un lien vers  
`http://www.monsite.fr/projet/images/new.gif`
  - le navigateur client reconstruit l'URL absolue pour faire la requête
  - la balise HTML `<BASE href="url">` permet de positionner la racine pour les URLs relatives du document contenant cette balise

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

253

253

## Vision côté client

- Le Web est un ensemble de pages (documents) pouvant contenir des liens vers d'autres pages n'importe où dans le monde
- Consultation des pages via un navigateur
- L'utilisateur suit ces liens par simple click --> notion d'hypertexte (information répartie)
- Le navigateur (*browser*)
  - analyse l'URL demandée
  - demande au DNS l'adresse IP du site distant
  - établit une connexion TCP vers le numéro de port de l'URL (80 par défaut)
  - formule la requête au serveur

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

254

254

## Vision côté client

- Le navigateur (*browser*)
  - va rechercher la page demandée
  - interprète les commandes de formatage et de mise en forme (police, gras, couleurs...)
  - va rechercher et affiche des images
  - animation (code JavaScript, gifs...)
  - affiche la page correctement formatée
- Paramétrage à plusieurs niveaux
  - valeurs par défaut du navigateur
  - valeurs fixées dans le document
  - préférences de l'utilisateur (navigateur)
  - exemples : couleur des liens (visités ou non), du texte, fond de la page, polices...

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

255

255

## Vision côté serveur

- Le serveur est en permanence à l'écoute des requêtes formulées par les clients (qui peuvent être très nombreux !)
- Il vérifie la validité de la requête...
  - adresse correcte (URL)
  - client autorisé à accéder au document
- ... et y répond : envoi du texte, des images, du code à exécuter sur le client, d'un message d'erreur, d'une demande d'authentification, ...
- Il peut exécuter un programme localement qui va générer une réponse HTML (pages dynamiques)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

256

256

## Pourquoi des formulaires ?

- Apporte de l'inter-activité avec l'utilisateur en proposant des zones de dialogue : un formulaire n'est qu'une interface de saisie !
- Selon les choix de l'utilisateur, il faut y associer un traitement
  - sur le client avec JavaScript par exemple
  - sur le serveur par l'intermédiaire de CGI, PHP, ...
- Exemples typiques d'utilisation de formulaire
  - commandes, devis via Internet
  - moteurs de recherche
  - interactions avec une base de données

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

257

257

## Principe du formulaire

- On décrit à l'aide de balises HTML les différents champs de saisie
- Chaque zone est identifiée par un nom symbolique auquel sera associée une valeur par l'utilisateur
- Quand le formulaire est soumis, les couples (nom/valeur) de toutes les zones sont transmis dans la requête HTTP au serveur
- A chaque zone de saisie peut être associé un traitement sur le client par l'intermédiaire d'un événement JavaScript

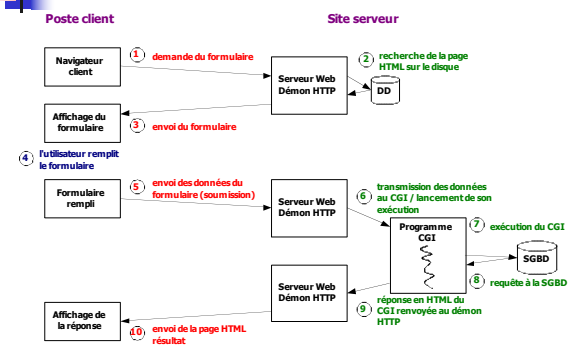
Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

258

258

## Le client est passif

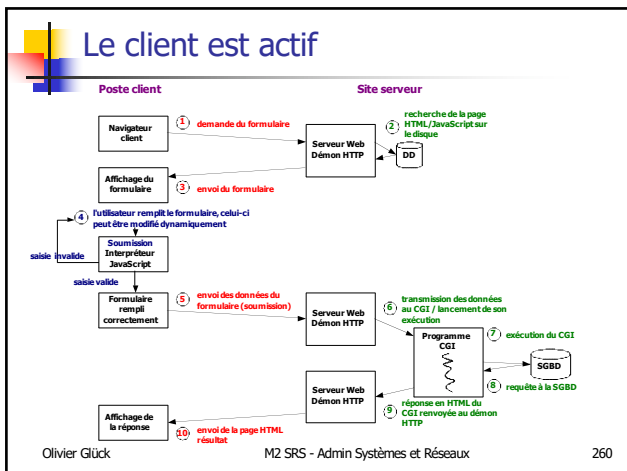


Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

259

259



260

## Caractéristiques de HTTP

- HTTP : Hyper Text Transfer Protocol
- Protocole régissant le dialogue entre des clients Web et un serveur (c'est le langage du Web !)
- Fonctionnement en mode Client/Serveur
- Une transaction HTTP contient
  - le type de la requête ou de la réponse (commande HTTP)
  - un en-tête
  - une ligne vide
  - un contenu (parfois vide)
- Très peu de type de requêtes/réponses
- Port standard : 80

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 261

261

## Une transaction typique (1)

- 1 - le client contacte le serveur pour demander le document index.html  
GET /~ogluck/index2.html HTTP/1.1
- 2 - le client envoie des informations d'en-tête pour informer le serveur de sa configuration et des documents qu'il accepte  
User-Agent: Mozilla/4.0 (compatible;MSIE 6.0;Windows NT 5.1)  
Host: www710.univ-lyon1.fr  
Accept: image/gif, image/jpeg, \*/\*
- 3 - le client envoie une ligne vide (fin de l'en-tête) et un contenu vide dans cet exemple

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 262

262

## Une transaction typique (2)

- 4 - le serveur répond en commençant par indiquer par un code, l'état de la requête  
HTTP/1.1 200 OK
- 5 - le serveur envoie un en-tête qui donne des informations sur lui-même et le document demandé  
Date: Sun, 23 May 2004 17:46:01 GMT  
Server: Apache/1.3.28 (Debian GNU/Linux) PHP/3.0.18  
Last-Modified: Sun, 23 May 2004 17:42:12 GMT  
Content-Length: 90  
Content-Type: text/html; charset=iso-8859-1
- 6 - puis une ligne vide (fin de l'en-tête) et le contenu du document si la requête a réussi

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 263

263

## Une transaction typique (3)

```

ogluck@lima:~$ telnet localhost 80
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
GET /~ogluck/index2.html HTTP/1.1
Host: localhost
Accept: */*

HTTP/1.1 200 OK
Date: Sun, 23 May 2004 17:46:01 GMT
Server: Apache/1.3.28 (Debian GNU/Linux) PHP/3.0.18
Last-Modified: Sun, 23 May 2004 17:42:12 GMT
ETag: "a805a-5a-40b0e274"
Accept-Ranges: bytes
Content-Length: 90
Content-Type: text/html; charset=iso-8859-1

<html><head><title>
index2.html
</title></head><body>
<h1>Bienvenue !</h1>
</body></html>
Connection closed by foreign host.
ogluck@lima:~$

```

Olivier Glück 264

264

## Format des requêtes/réponses

- Format des requêtes
  - type de la requête (METHOD, URL, version HTTP)
  - en-tête
  - une ligne vide
  - un contenu éventuel
- Format des réponses
  - un code de réponse (version HTTP, code, description)
  - en-tête
  - une ligne vide
  - le contenu de la réponse

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 265

265

## Durée de vie des connexions

- HTTP 1.0 (RFC 1945)
  - dès que le serveur a répondu à une requête, il ferme la connexion HTTP
- HTTP 1.1 (RFC 2068)
  - par défaut, la connexion est maintenue tant que le serveur ou le client ne décide pas de la fermer (`Connection: close`)
- HTTP est un protocole **sans état**
  - aucune information n'est conservée entre deux connexions
  - permet au serveur HTTP de servir plus de clients en un temps donné (gestion légère des transactions)
  - pour conserver des informations entre deux transactions, il faut utiliser un *cookie*, des champs cachés d'un formulaire, ...

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 266

266

## Cookies

- Moyen pour le serveur de stocker des informations chez le client pour palier au caractère sans état du protocole HTTP
- Cookie = une chaîne de caractères url-encodée de 4ko max stockée sur le disque dur du client
- Informations associées à un ensemble d'URL qui sont envoyées lors de toute requête vers l'une de ces URL
- Les *cookies* permettent de
  - propager un code d'accès (évite une authentification lors de chaque requête)
  - identification dans une base de données
  - fournir des éléments statistiques au serveur (compteurs de pages visitées, ...)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 267

267

## Installation d'un Cookie sur le client

- Directive Set-Cookie dans l'en-tête de la réponse HTTP (envoyé lors de la première connexion)

Set-Cookie: nom=valeur; expires=date; path=chemin\_accès; domain=nom\_domaine; secure

- le couple nom/valeur est le contenu du cookie (seul champ obligatoire), sans espace ; et ,
- le cookie devient invalide après la date indiquée
- path=/pub signifie que le cookie est valable pour toutes les requêtes dont l'URL contient /pub
- domain indique le nom de domaine (associé au serveur) pour lequel le cookie est valable
- secure : le cookie n'est valable que lors d'une connexion sécurisée

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 268

268

## Utilisation d'un Cookie par le client

- Chaque fois qu'un client va effectuer une requête, il vérifie dans sa liste de *cookies* s'il y en a un qui est associé à cette requête
- Si c'est le cas, le client utilise la directive Cookie dans l'en-tête de la requête HTTP

Cookie: nom1=valeur1; nom2=valeur2; ...

- Le serveur peut insérer plusieurs directives Set-Cookie
- Dans la première spécification des *cookies* :
  - un client peut stocker un maximum de 300 *cookies*
  - un maximum de 20 *cookies* par domaine est permis
  - la taille d'un *cookie* est limitée à 4Ko

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 269

269

## Différentes versions de HTTP (1)

- Version d'origine : HTTP 0.9
  - Une seule méthode : GET
  - Pas d'en-têtes
  - Une requête = une connexion TCP
- Amélioration en 2 étapes
  - HTTP 1.0 :
    - introduction des en-têtes (échange de "méta" info)
    - nouvelles possibilités : utilisation de caches, méthodes d'authentification, ...
  - HTTP 1.1 :
    - mode **connexions persistantes** par défaut
    - introduction des serveurs virtuels -> la directive `Host` dans la requête est nécessaire

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 270

270

## Différentes versions de HTTP (2)

- Intérêt des connexions persistantes
  - exemple d'une page d'accueil avec 5 images
  - HTTP 0.9 : 6 connexions/déconnexions TCP/IP
  - HTTP 1.1 : 1 seule connexion TCP/IP
- Intérêt d'un cache - amélioration des performances
  - les pages qui sont le plus souvent demandées sont conservées dans un cache
  - -> soulage le réseau
  - -> accès plus rapide
  - peut être utilisé localement ou par l'intermédiaire d'un serveur relais (*proxy*)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 271

271

## Connexions persistantes

- Non-persistante
  - HTTP/1.0
  - le serveur interprète les requêtes, répond et ferme la connexion TCP
  - 2 RTTs sont nécessaires pour lire chaque objet
  - chaque transfert doit supporter le *slow-start*
  - exemple page contenant :
    - 1 fichier HTML
    - 10 images JPEG
- Persistante
  - par défaut dans HTTP/1.1
  - une seule connexion TCP est ouverte vers le serveur
  - le client envoie la requête de tous les objets requis dès qu'ils sont référencés dans le code HTML
  - moins de RTTs et moins de *slow-start*
  - deux versions : avec/sans pipeline

Mais la plupart des navigateurs de version 1.0 utilisent des connexions parallèles

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 272

272

## Proxy

Mise en cache hiérarchique

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 273

273

## Intérêt du cache Web

- Hypothèse : le cache est proche du client
- Réduction du temps de réponse
- Réduction du débit vers les serveurs distants

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 274

274

## Les requêtes du client

- Rappel : Format d'une requête
  - une commande HTTP (METHOD), une URL qui identifie la ressource demandée, la version de HTTP
  - l'en-tête et une ligne vide
  - éventuellement un contenu (corps de la requête)
- Méthode GET
- Méthode POST
- Méthode HEAD
- D'autres méthodes qui ne sont pas souvent supportées par les serveurs

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 275

275

## La méthode GET

- La méthode standard de requête d'un document
  - récupérer un fichier, une image, ...
  - activer un script CGI en lui transmettant des données
- Le contenu de la requête est toujours vide
- Le serveur répond avec une ligne décrivant l'état de la requête, un en-tête et le contenu demandé
- Si la requête échoue, le contenu de la réponse décrit la raison de l'échec (fichier non présent, non autorisé, ...)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 276

276

## La méthode GET et les CGI

- Comme le contenu d'une requête GET est vide, les données du formulaire sont transmises via l'URL après un ?
- Les champs sont séparés par un &

`GET /cgi-bin/prog.cgi?email=toto@site.fr&pass=toto&s=login HTTP/1.1`

- Ici, trois champs du formulaire sont transmis dans la requête
- Le mot de passe est transmis en clair !
- Permet de conserver dans un *bookmark* les données saisies dans le formulaire
- L'URL a une taille limitée (4Ko)

Olivier Glück M2 SRS - Admin Systèmes et Réseaux 277

277

## La méthode POST

- Elle permet de transmettre des données au serveur dans le corps de la requête
- Exemple

```
POST /cgi-bin/prog.cgi HTTP/1.1
User-Agent: Mozilla/4.0 (compatible;MSIE 6.0;Windows NT 5.1)
Host: localhost
Accept: */*
Content-type: application/x-www-form-urlencoded
Content-length: 36

email=toto@site.fr&pass=toto&s=login
```
- Le mot de passe est toujours transmis en clair !

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

278

278

## La méthode HEAD (1)

- Identique à GET mais permet uniquement de récupérer l'en-tête relatif à un document
- Permet de récupérer
  - la date de dernière modification du document (important pour les caches, JavaScript)
  - la taille du document (estimation du temps d'arrivée du document)
  - le type du document (le client peut sélectionner le type de documents qu'il accepte)
  - le type du serveur (permet de faire des requêtes spécifiques selon le type du serveur)
- Remarque : le serveur ne fournit pas nécessairement toutes ces informations !

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

279

279

## La méthode HEAD (2)

```
xterm
ogluck@liwa:~$ telnet localhost 80
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
HEAD /"ogluck/index2.html HTTP/1.0
Accept: */*

HTTP/1.1 200 OK
Date: Sun, 23 May 2004 18:14:14 GMT
Server: Apache/1.3.28 (Debian GNU/Linux) PHP/3.0.18
Last-Modified: Sun, 23 May 2004 17:42:12 GMT
ETag: "a805a-5a-40b0e274"
Accept-Ranges: bytes
Content-Length: 90
Connection: close
Content-Type: text/html; charset=iso-8859-1

Connection closed by foreign host.
ogluck@liwa:~$
```

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

280

280

## Autres requêtes clientes

- PUT : permet de stocker le corps de la requête sur le serveur à l'URL spécifiée
- DELETE : suppression du document spécifié par l'URL
- OPTIONS : renvoie la liste des méthodes autorisées par le serveur
- TRACE : la corps de la requête entrante est renvoyée au client - utilisé pour faire du debug
- ...

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

281

281

## Les réponses du serveur

- Les codes de réponse
  - trois parties : version HTTP, code de statut, description textuelle du code

```
HTTP/1.1 200 OK
HTTP/1.1 404 Not Found
```
  - code=entier sur 3 chiffres classé selon des catégories
    - 100-199 : message d'information
    - 200-299 : succès de la requête cliente
    - 300-399 : la requête n'est pas directement serviable, le client doit préciser certaines choses
    - 400-499 : échec de la requête qui incombe au client
    - 500-599 : échec de la requête qui incombe au serveur (par ex. erreur d'exécution d'un CGI)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

282

282

## Quelques en-têtes de requêtes

- Identification du client
  - From (adresse mail du client), Host (serveur, **obligatoire en HTTP1.1**), Referer (URL d'où l'on vient), User-Agent
- Préférences du client
  - Accept (liste des types MIME acceptés), Accept-Encoding (compress|gzip|...), Accept-Langage, Accept-Charset
- Information pour le serveur
  - Autorization (username:passwd encodé en base64), Cookie
- Conditions sur la réponse
  - If-Modified-Since (utile pour les caches), If-Unmodified-Since, If-Match (Etag)

Olivier Glück

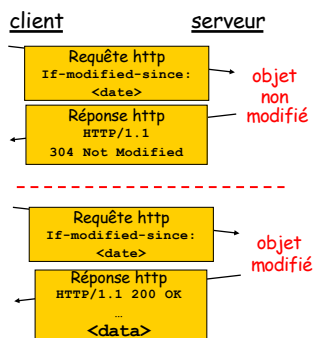
M2 SRS - Admin Systèmes et Réseaux

283

283

## Quelques en-têtes de requêtes

- Objectif : ne pas envoyer un objet que le client a déjà dans son cache
- Problème : les objets contenus dans le cache peuvent être obsolètes
- Le client spécifie la date de la copie cachée dans la requête http  
If-modified-since: <date>
- la réponse du serveur est vide si la copie cachée est à jour



Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

284

284

## Quelques en-têtes de réponses

- Informations sur le contenu du document  
Content-Type (type MIME du document),  
Content-Length (barre de progression du chargement),  
Content-Encoding, Content-Location,  
Content-Language
- Informations sur le document  
Last-Modified (date de dernière modification),  
Allow (méthodes autorisées pour ce document),  
Expires (date d'expiration du document)
- En-tête générales  
Date (date de la requête), Server (type du serveur)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

285

285

## Transfert par morceaux en HTTP/1.1

- La réponse peut être envoyée en plusieurs morceaux (dans le cas des CGI par exemple car le serveur ne peut pas toujours déterminer la longueur totale de la réponse)  
Transfer-Encoding: Chunked
- Chaque morceau est constitué d'une ligne comportant la taille du morceau en hexadécimal puis des données
- Après les morceaux, une ligne contenant 0 et éventuellement des en-têtes supplémentaires

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

286

286

## Les types MIME

- MIME : *Multi-purpose Internet Mail Extensions*
- Permet l'échange de fichiers multimédias entre machines quelconques en spécifiant le type du fichier
- Les commandes MIME ont été intégrées dans HTTP1.0
- Un type MIME est composé
  - d'un type général (text, image, audio, video, application...)
  - et d'un sous-type (image/gif, image/jpeg, application/pdf, application/rtf, text/plain, text/html)
- En perpétuelle évolution
- La machine cliente doit ensuite associer l'exécution d'une application à chaque type MIME
- Le serveur positionne Content-type à partir de l'extension du document demandé (/etc/mime.types)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

287

287

## CGI - Common Gateway Interface

- Interface de base qui définit la communication entre le serveur HTTP et un programme d'application
- CGI spécifie comment des navigateurs clients peuvent communiquer avec des programmes qui s'exécutent sur le serveur Web et qui génèrent des pages HTML dynamiques **créées à la volée** à partir du résultat des exécutions

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

288

288

## Qu'est ce qu'un programme CGI ?

- Un programme
  - qui s'exécute sur la machine hébergeant le serveur HTTP
  - en langage compilé (binaire) ou interprété (script)
  - qui permet de
    - recupérer les données du formulaire à l'aide d'un *parser* : pour chaque champ, un couple NAME/VALUE est transmis au serveur
    - effectuer des traitements sur le serveur
      - lecture/écriture dans une base de données
      - stockage d'une info (compteurs, identifiant de connexion, ...)
      - recherche d'une info
      - pied de page automatique (ex: date de dernière modification)
    - générer un résultat qui est renvoyé au client
      - page HTML, image, document postscript, ...

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

289

289



## Avantages/inconvénients

- Puissant mais dangereux
  - permet d'exécuter tout et n'importe quoi par le démon HTTP du serveur
- Un CGI doit s'exécuter rapidement
  - risque de surcharge du serveur
  - utilisateurs impatient : pendant que le CGI s'exécute, le client attend la réponse sans savoir pourquoi elle n'arrive pas...
  - possibilité d'envoyer dès le début de l'exécution une page qui permet d'indiquer à l'utilisateur que le résultat va arriver

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

290

290

## Un premier exemple (1)

```
#!/bin/sh
Date.cgi
echo 'Content-type: text/html'
echo ''
#Création du corps du document
echo '<HTML><HEAD><TITLE>'
echo 'Date.cgi'
echo '</TITLE></HEAD><BODY>'
echo '<H1>Date sur le serveur</H1>'
echo -n "On est le `date +%D`, il est "
echo "`date +%H`h `date +%M`m"
echo '</BODY></HTML>'
```

```
ogluck@lima:~/public_html/cgi-
bin$./Date.cgi
Content-type: text/html

<HTML><HEAD><TITLE>
Date.cgi
</TITLE></HEAD><BODY>
<H1>Date sur le serveur</H1>
On est le 11/07/03, il est 11h
30m
</BODY></HTML>
```

Source du programme CGI

Exécution du CGI sur le serveur

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

291

291

## Un premier exemple (2)

Exécution du CGI depuis le client



Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

292

292

## Un premier exemple (3)

- Ce programme CGI n'utilise aucune donnée en provenance du client
- Il récupère simplement la date sur le serveur et affiche sur sa **sortie standard** le code d'une page HTML minimale contenant la date et l'heure
- La ligne "Content-type: text/html" est une information destinée au serveur pour la construction de l'en-tête HTTP constituant la réponse renvoyée au client (ici, il s'agit d'indiquer que le type des données générées par le CGI est une suite de commandes HTML)

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

293

293

## Méthodes GET/POST (1)

- Voici le code d'un petit script CGI en shell

```
#!/bin/sh
Get_Post.cgi
echo 'Content-type: text/plain'
echo ''
echo "QS=$QUERY_STRING"
read DATA
echo "Data=$DATA"
```
- Les résultats de l'exécution avec la méthode GET puis POST sont montrés dans les deux transparents suivants

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

294

294

## Méthodes GET/POST (2)

```
ogluck@lima:~$ telnet localhost 80
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
GET /"ogluck/cgi-bin/Get_Post.cgi?email=toto@site.fr&pass=toto&=login HTTP/1.1
Host: localhost
Accept: */*

HTTP/1.1 200 OK
Date: Sun, 23 May 2004 18:25:26 GMT
Server: Apache/1.3.28 (Debian GNU/Linux) PHP/3.0.18
Transfer-Encoding: chunked
Content-Type: text/plain; charset=iso-8859-1

2e
Data=email=toto@site.fr&pass=toto&=login
0

Connection closed by foreign host.
ogluck@lima:~$
```

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

295

295

## Méthodes GET/POST (3)

```
xterm
ogluck@lina:~$ telnet localhost 80
Trying 127.0.0.1...
Connected to localhost.
Escape character is '^]'.
POST /"/ogluck/cgi-bin/Get_Post.cgi HTTP/1.1
Accept: */*
Host: localhost
Content-type: application/x-www-form-urlencoded
Content-length: 36

email=toto@site.fr&pass=toto&s=login
HTTP/1.1 200 OK
Date: Sun, 23 May 2004 18:29:52 GMT
Server: Apache/1.3.23 (Debian GNU/Linux) PHP/3.0.18
Transfer-Encoding: chunked
Content-Type: text/plain; charset=iso-8859-1

4
QS=
2a
Data=email=toto@site.fr&pass=toto&s=login
0
Connection closed by foreign host.
ogluck@lina:~$
```

Olivier Glück

296

296

## Méthodes GET/POST (4)

- Avec la méthode GET
  - les données relatives aux champs du formulaire sont transmises via l'URL (dans le type de la requête)
  - le programme CGI les récupère dans la variable d'environnement `QUERY_STRING`
  - il est possible de cliquer sur "Actualiser" pour retransmettre les données et de définir un *bookmark*
- Avec la méthode POST
  - les données relatives aux champs du formulaire sont transmises dans le corps de la requête HTTP
  - `Content-type` et `Content-length` sont positionnés
  - le programme CGI les récupère sur l'entrée standard
  - "Actualiser" et *bookmark* impossibles, données du formulaire non visibles dans les logs du serveur

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

297

297

## Méthodes GET/POST (5)

### Formulaire

### Méthode GET

Adresse [http://lina/cg-bin/Get\\_Post.cgi?email=toto@site.fr&pass=toto&s=login](http://lina/cg-bin/Get_Post.cgi?email=toto@site.fr&pass=toto&s=login)

QS=email=toto@site.fr&pass=toto&s=login  
Data=

Adresse [http://lina/cg-bin/Get\\_Post.cgi](http://lina/cg-bin/Get_Post.cgi)

QS=  
Data=email=toto@site.fr&pass=toto&s=login

### Méthode POST

### Méthode POST et "Actualiser"

Adresse [http://lina/cg-bin/Get\\_Post.cgi](http://lina/cg-bin/Get_Post.cgi)

Microsoft Internet Explorer

La page ne peut pas être actualisée sans le renvoi d'informations.  
Cliquez sur **Recommencer** pour renvoyer les informations ou sur **Annuler** pour revenir à la page que vous essayez de consulter.

Recommencer Annuler

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

298

298

## Format URL encodé (1)

- Nécessité de coder les données de l'URL (méthode GET) sur le client pour construire la chaîne CGI pour respecter la RFC 2396 qui spécifie la syntaxe des URL
- Les caractères non-alphanumériques sont remplacés par `%xx` (xx=code ASCII du caractère en hexadécimal)
- Les caractères `;` `?` `:` `@` `&` `=` `+` `$` et `,` sont réservés
  - `?` : début de `QUERY_STRING`
  - `&` : séparateur de champ
  - `=` : séparation entre le nom du champ et sa valeur
- Les espaces sont remplacés par `+`

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

299

299

## Format URL encodé (2)

- Format de la chaîne CGI  
`nom_champ1=valeur1&nom_champ2=valeur2&...`
- Cas des champs à valeurs multiples
  - exemple : listes à sélection multiples  
`nom_liste=valeur1&nom_liste=valeur2&...`
- La chaîne CGI
  - elle est construite par le client au format *URL-encoded* quand la requête est postée
  - elle est transmise au CGI tel quel via la variable d'environnement `QUERY_STRING` avec la méthode GET
  - elle est transmise au CGI tel quel via l'entrée standard avec la méthode POST

Olivier Glück

M2 SRS - Admin Systèmes et Réseaux

300

300