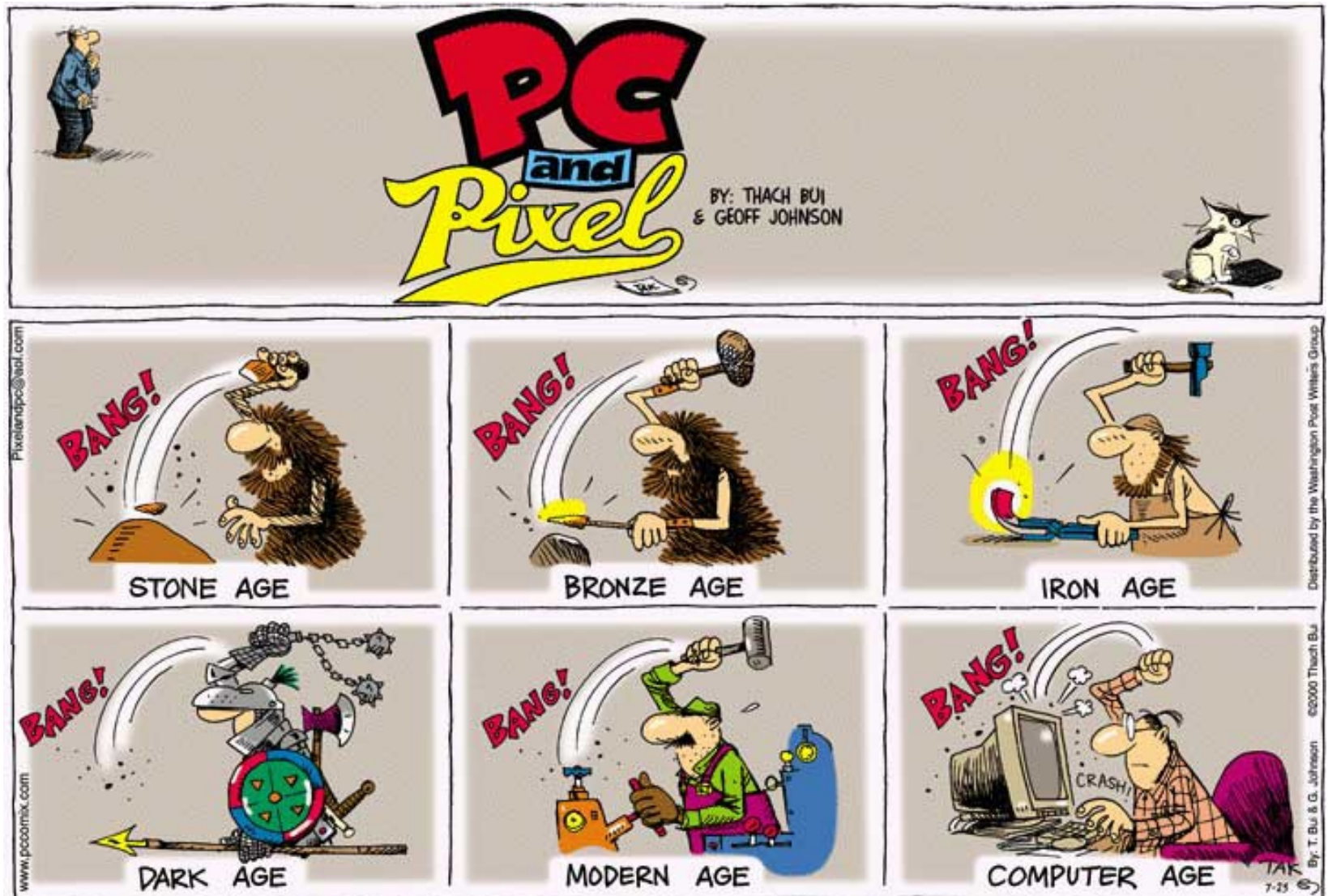


Niveau 7

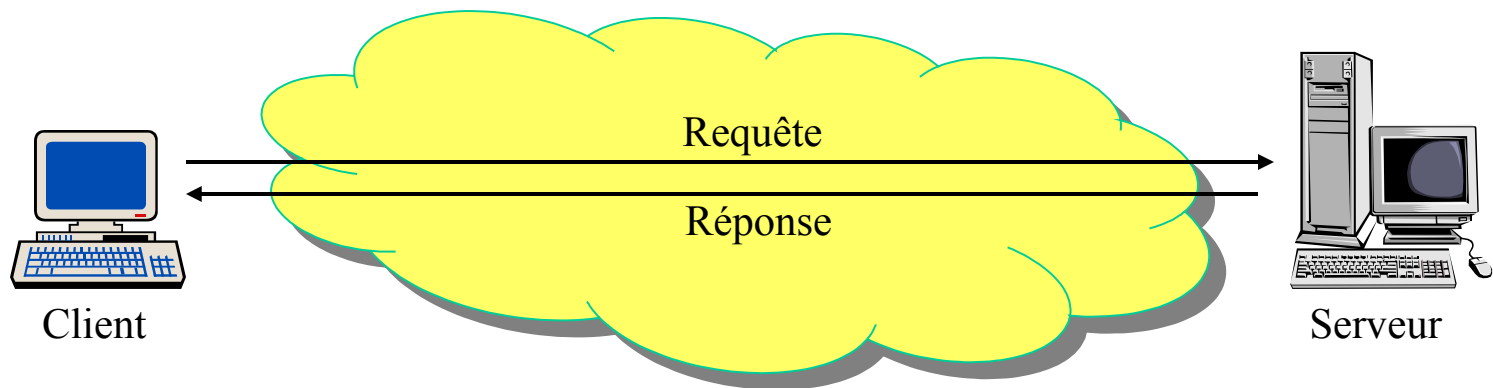
Applications: Telnet, SSH, FTP,
TFTP, HTTP, SNMP, Mail, (DNS),
NTP, P2P (+ VoIP)...

Applications – Utilisateur !



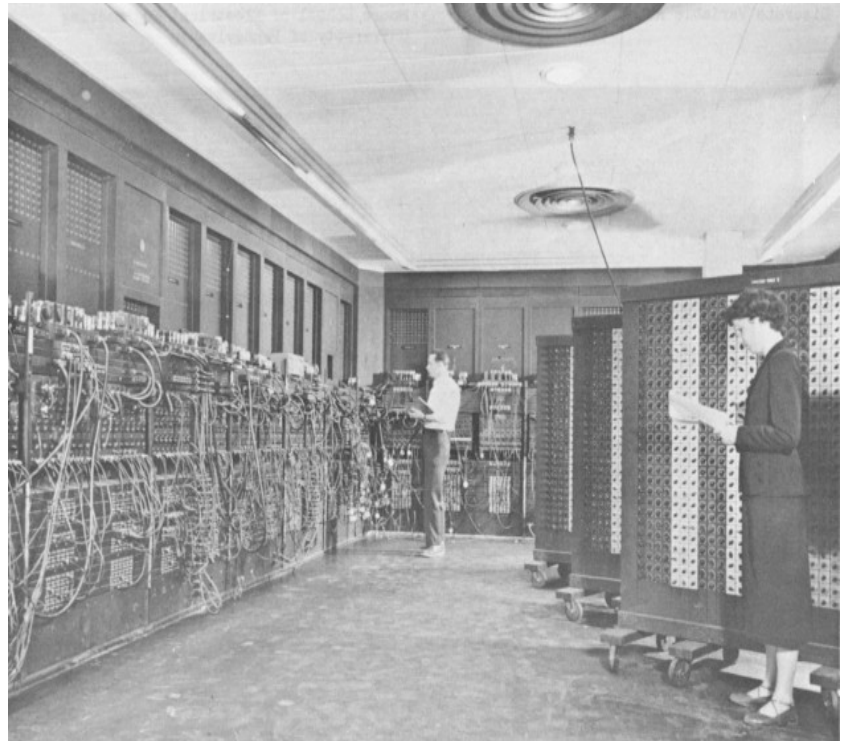
Applications – Généralités 1

- Les applications (couche 7) sont souvent basées sur une architecture client – serveur
- Ce sont elles dont les utilisateurs ont besoin !!!
- Ne pas confondre l'application (interface, programme) et le protocole (règles) !



Le Cloud Computing

- Très à la mode.
- On revient à un système centralisé, comparable à ce qui existait en... 1960 !
- Mais cela a un peu évolué !



Situation 1960

- Terminaux (simplissimes)
- Lignes à 9600 bps
- Mainframe (une machine centrale)

Situation 2010

- PC (Windows / Mac / Linux)
- Connexion Internet (p. ex. ADSL : 4Mbps)
- Serveurs Web (une ou plusieurs machines)

MORALITE :
retour à une architecture
d'il y a 50 ans !!!

Le Cloud Computing

- Avantages :
 - On utilise des spécialistes pour pas cher (économie d'échelle)
 - Les updates / upgrades sont faits automatiquement
 - Pas de gestion de l'outil
- Désavantages :
 - Dépendance vis à vis de la connexion Internet
 - Plus difficile d'avoir du 'custom development'

Cloud Computing – Exemples

- Mail (Gmail)
- Backup
- CRM
- Comptabilité
- Central téléphonique
- Gestion de stock, de tickets, de mailing list
- De plus en plus d'applications sur PDA / smartphones
- Etc.....

Applications – Généralités 2

- Basé sur la notion de porte (TCP / UDP)
- Processus dans le serveur (daemon / process)
- Plusieurs processus en parallèle sur un serveur
- Chacun sa porte (éventuellement configurable) !!!
- Grande différence entre les standards (de juro) et les applications (de facto)

Telnet

- Utilisé pour se connecter à une machine à partir d'une autre.
- TCP port 23
- Aucune sécurité
- Basé sur la notion de Virtual Terminal
- Une des premières applications (avec FTP) !!!
- Protocoles similaires: rlogin, rsh, rcp

Telnet – Détails 1

- RFC 854
- Options négociées (RFC 2400, en évolution)
- Négociation des commandes:
 - DO L'émetteur veut que le récepteur fasse quelque chose
 - DON'T L'émetteur ne veut pas que le récepteur fasse quelque chose
 - WILL L'émetteur veut faire quelque chose
 - WON'T L'émetteur ne veut pas faire quelque chose

Telnet – Options

Protocol	Name		Number	State
• TOPT-BIN	Binary Transmission	0	Std	
• TOPT-ECHO	Echo	1	Std	
• TOPT-RECN	Reconnection		2	Prop
• TOPT-SUPP	Suppress Go Ahead	3	Std	
• TOPT-APRX	Approx Message Size Negotiation	4	Prop	
• TOPT-STAT	Status	5	Std	
• TOPT-TIM	Timing Mark	6	Std	
• TOPT-REM	Remote Controlled Trans and Echo	7	Prop	
• TOPT-OLW	Output Line Width	8	Prop	
• TOPT-OPS	Output Page Size	9	Prop	
• TOPT-OCRD	Output Carriage-Return Disposition	10	Prop	
• TOPT-OHT	Output Horizontal Tabstops		11	Prop
• TOPT-OHTD	Output Horizontal Tab Disposition	12	Prop	
• TOPT-OFD	Output Formfeed Disposition	13	Prop	
• TOPT-OVT	Output Vertical Tabstops	14	Prop	
• TOPT-OVTD	Output Vertical Tab Disposition	15	Prop	
• TOPT-OLD	Output Linefeed Disposition		16	Prop
• TOPT-EXT	Extended ASCII	17	Prop	
• TOPT-LOGO	Logout	18	Prop	
• TOPT-BYTE	Byte Macro	19	Prop	
• TOPT-DATA	Data Entry Terminal	20	Prop	
•				

Telnet – Commandes utilisateur

**Pour avoir le
prompt Telnet
à partir d'une
session en cours:
'^I'**

```
telnet> ?
```

Commands may be abbreviated. Commands are:

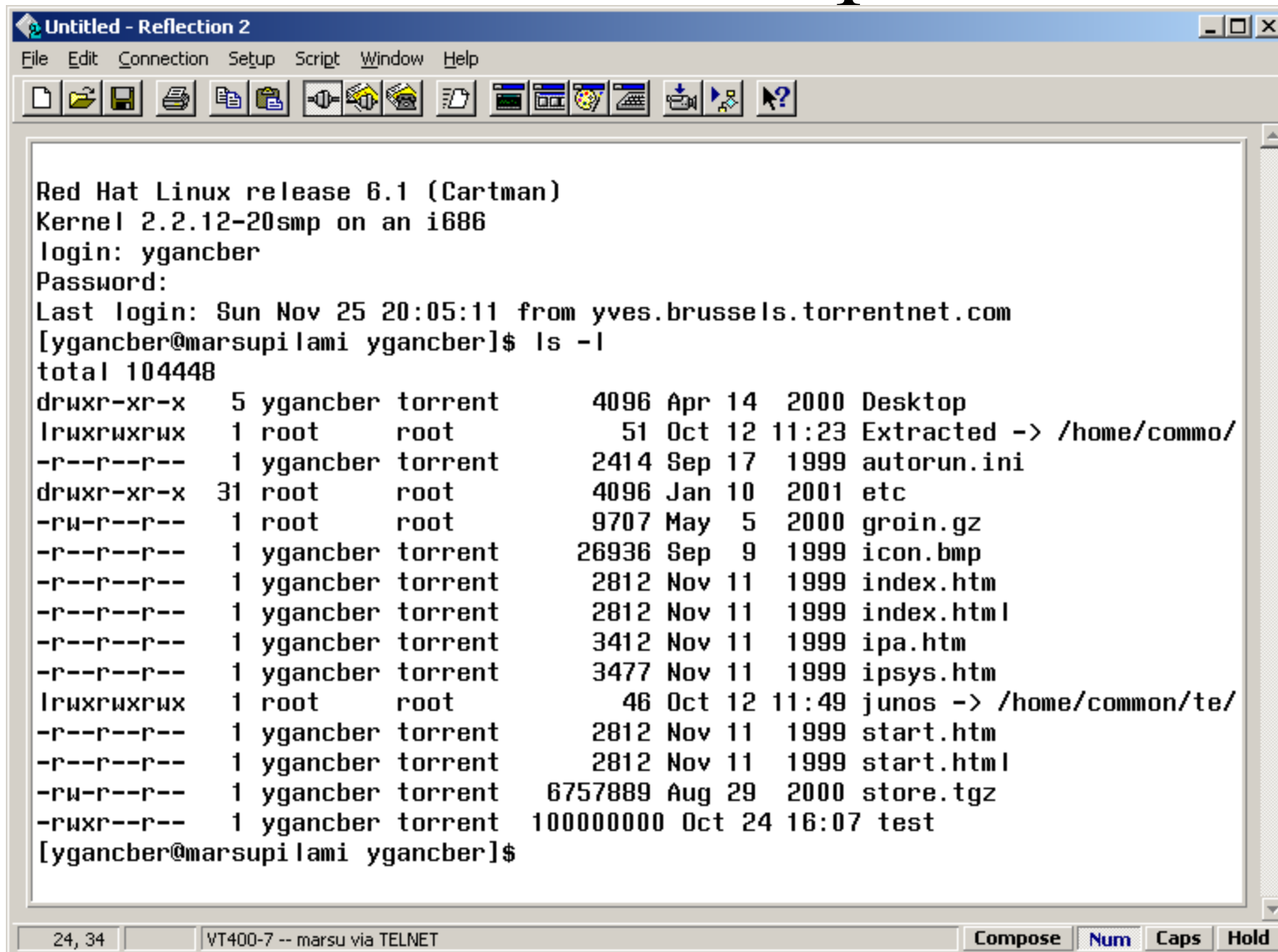
close	close current connection
logout	forcibly logout remote user and close the connection
display	display operating parameters
mode	try to enter line or character mode ('mode ?' for more)
open	connect to a site
quit	exit telnet
send	transmit special characters ('send ?' for more)
set	set operating parameters ('set ?' for more)
unset	unset operating parameters ('unset ?' for more)
status	print status information
toggle	toggle operating parameters ('toggle ?' for more)
slc	change state of special charaters ('slc ?' for more)
z	suspend telnet
!	invoke a subshell
environ	change environment variables ('environ ?' for more)
?	print help information

Telnet – Exemple 1

```
Command Prompt - telnet marsu

Red Hat Linux release 6.1 (Cartman)
Kernel 2.2.12-20smp on an i686
login: ygancher
Password:
[ygancher@marsupilami ygancher]$ ls -l
total 104448
drwxr-xr-x  5 ygancher torrent      4096 Apr 14  2000 Desktop
lrwxrwxrwx  1 root      root         51 Oct 12  11:23 Extracted -> /home/common
/tech/Juniper/Imagefiles/3.4/Extracted/
-r--r--r--  1 ygancher torrent     2414 Sep 17  1999 autorun.ini
drwxr-xr-x  31 root      root      4096 Jan 10  2001 etc
-rw-r--r--  1 root      root     9707 May  5  2000 groin.gz
-r--r--r--  1 ygancher torrent    26936 Sep  9  1999 icon.bmp
-r--r--r--  1 ygancher torrent     2812 Nov 11  1999 index.htm
-r--r--r--  1 ygancher torrent     2812 Nov 11  1999 index.html
-r--r--r--  1 ygancher torrent     3412 Nov 11  1999 ipa.htm
-r--r--r--  1 ygancher torrent     3477 Nov 11  1999 ipsys.htm
lrwxrwxrwx  1 root      root         46 Oct 12  11:49 junos -> /home/common/tec
h/Lab/PC-router/Olive/3.4R3.2/
-r--r--r--  1 ygancher torrent     2812 Nov 11  1999 start.htm
-r--r--r--  1 ygancher torrent     2812 Nov 11  1999 start.html
-rw-r--r--  1 ygancher torrent    6757889 Aug 29  2000 store.tgz
-rwxr--r--  1 ygancher torrent 1000000000 Oct 24  16:07 test
[ygancher@marsupilami ygancher]$
```

Telnet – Exemple 2



```
Red Hat Linux release 6.1 (Cartman)
Kernel 2.2.12-20smp on an i686
login: ygancber
Password:
Last login: Sun Nov 25 20:05:11 from yves.brussels.torrentnet.com
[ygancber@marsupilami ygancber]$ ls -l
total 104448
drwxr-xr-x  5 ygancber torrent      4096 Apr 14  2000 Desktop
lrwxrwxrwx  1 root      root          51 Oct 12 11:23 Extracted -> /home/commo/
-r--r--r--  1 ygancber torrent    2414 Sep 17  1999 autorun.ini
drwxr-xr-x 31 root      root      4096 Jan 10  2001 etc
-rw-r--r--  1 root      root     9707 May  5  2000 groin.gz
-r--r--r--  1 ygancber torrent 26936 Sep  9  1999 icon.bmp
-r--r--r--  1 ygancber torrent  2812 Nov 11  1999 index.htm
-r--r--r--  1 ygancber torrent  2812 Nov 11  1999 index.html
-r--r--r--  1 ygancber torrent  3412 Nov 11  1999 ipa.htm
-r--r--r--  1 ygancber torrent  3477 Nov 11  1999 ipsys.htm
lrwxrwxrwx  1 root      root         46 Oct 12 11:49 junos -> /home/common/te/
-r--r--r--  1 ygancber torrent  2812 Nov 11  1999 start.htm
-r--r--r--  1 ygancber torrent  2812 Nov 11  1999 start.html
-rw-r--r--  1 ygancber torrent 6757889 Aug 29  2000 store.tgz
-rwxr--r--  1 ygancber torrent 100000000 Oct 24 16:07 test
[ygancber@marsupilami ygancber]$
```

Telnet – Example 3 (display)

```
telnet> display
will flush output when sending interrupt characters.
won't send interrupt characters in urgent mode.
won't skip reading of ~/.telnetrc file.
won't map carriage return on output.
will recognize certain control characters.
won't turn on socket level debugging.
won't print hexadecimal representation of network traffic.
won't print user readable output for "netdata".
won't show option processing.
won't print hexadecimal representation of terminal traffic.
```

```
echo      [^E]
escape    [^]]
rlogin    [off]
tracefile "(standard output)"
flushoutput [^O]
interrupt [^C]
quit      [^]
eof        [^D]
erase      [^?]
kill       [^U]
lnext      [^V]
susp       [^Z]
reprint    [^R]
worderase  [^W]
start      [^Q]
stop       [^S]
forw1      [off]
forw2      [off]
ayt        [^T]
```

```
WILL BINARY
DO ECHO
resp WILL_WONT ECHO: 1
want WONT ECHO
DO SUPPRESS GO AHEAD
DO STATUS
WILL TERMINAL TYPE
WILL NAWS
WILL TSPEED
WILL LFLOW
WILL LINEMODE
WILL XDISPLOC
resp WILL_WONT OLD-ENVIRON: 1
want WONT OLD-ENVIRON
WILL NEW-ENVIRON
```


SSH

- Secure SHell
- Utilisé pour se connecter à une machine à partir d'une autre
- TCP port 22
- Accès sécurisé (encryption)
- v2
- Vient de Unix

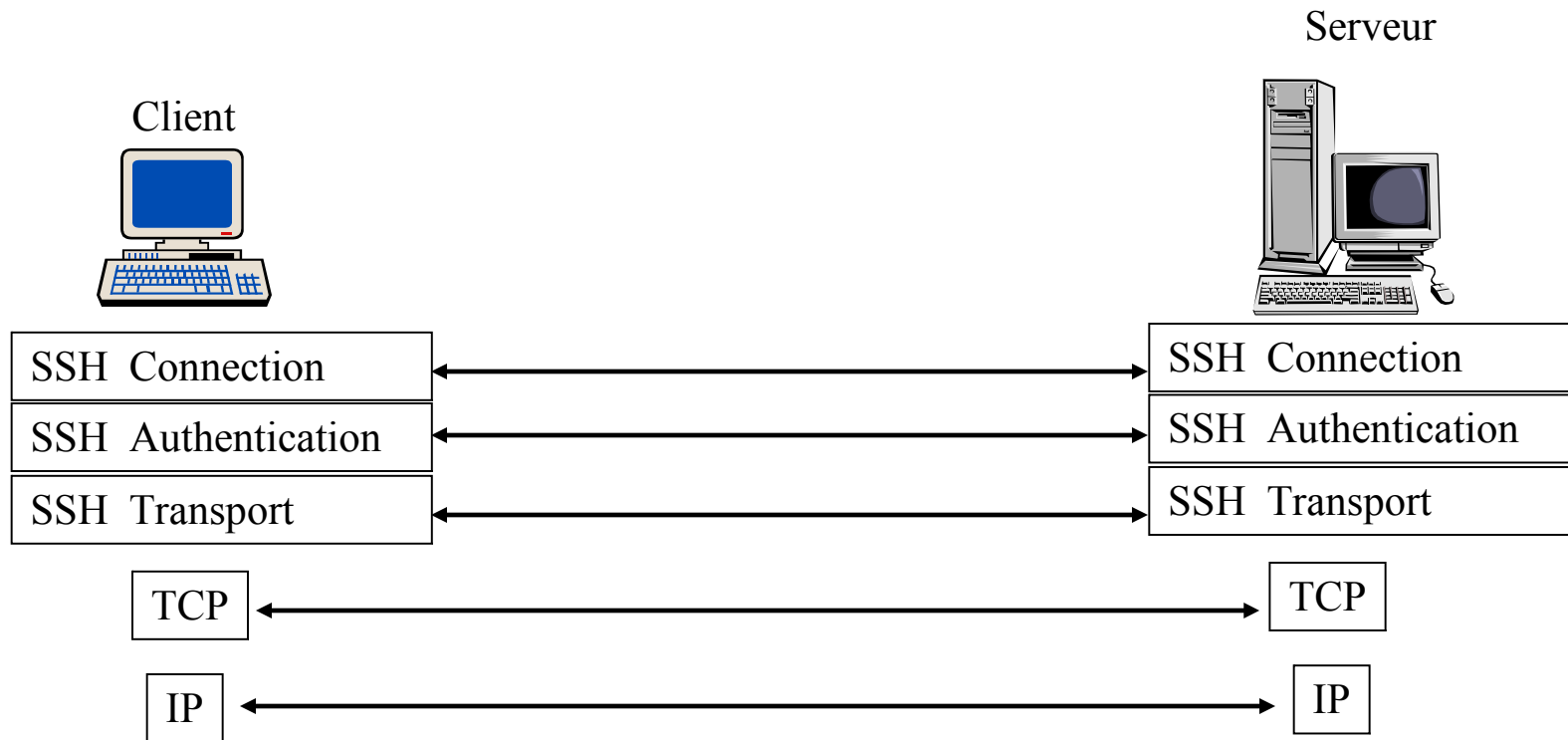
SSH– Détails 1

- Divers algorithmes d'encryption (voir plus loin): Propriétaire (v1), RSA, DES, 3DES, blowfish ...
- v2 plus riche et plus sécurisé que v1
- v3 en préparation
- Protège du 'sniffing' du réseau et des attaques 'man in the middle (MIM ou MITM)'

SSH v2 – Détails 2

- Basé sur trois sous-protocoles:
 - SSH Authentication protocol (authentification du serveur, négociation des paramètres, échange de clés (si nécessaire))
 - SSH Connection protocol (authentification de l'utilisateur, confidentialité)
 - SSH Transport Layer protocol (login interactifs, exécution à distance, mux de plusieurs sessions sur un canal)

SSH v2 – Détails 3



FTP

- File Transfer Protocol
- Utilisé pour transférer un (ou plusieurs) fichier d'une machine à l'autre
- TCP port 21
- Aucune sécurité
- Une des premières applications (avec Telnet) !!!

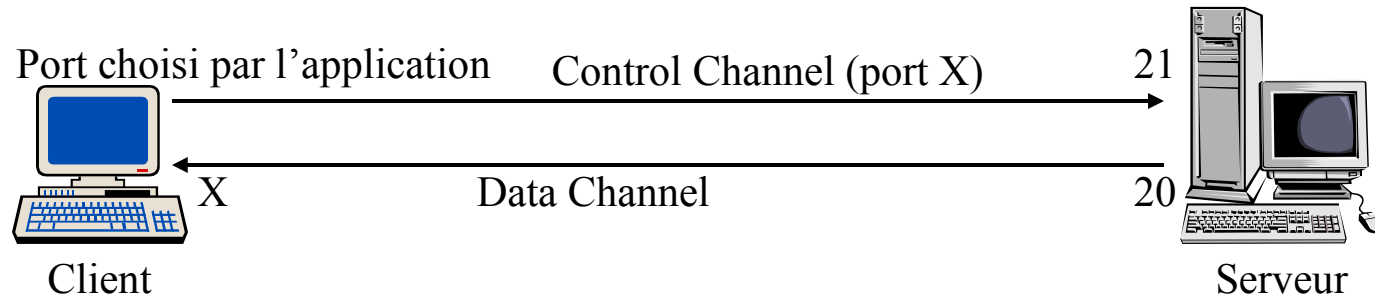
FTP – Détails 1

- RFC 959 (1985 !)
- Notion de login anonyme ('anonymous' user). Pas besoin d'être un utilisateur connu du serveur.
- Fonctionnement:
 - Login
 - Commande[s] utilisateur (à ne pas confondre avec les commandes du protocole)
 - Fin

FTP – Détails 2

- Fonctionnement de FTP:
 - Le client contacte le serveur pour établir une connexion
Il utilise le port 21 sur le serveur. C'est le 'Control Channel'
 - Il envoie une commande, PORT, au serveur, pour l'informer du port sur lequel le client s'attend à envoyer / recevoir des données
 - Le serveur crée une deuxième connexion (le 'Data Channel'), avec comme source port le 20 et comme destination port le numéro du port spécifié dans la commande PORT
 - C'est sur cette deuxième connexion, établie par le serveur, que le client enverra / recevra les données (d'où le nom de 'Data Channel')

FTP – Détails 3



FTP – Commandes classiques

- ascii – bin
- cd – lcd
- dir – ls
- hash
- pwd – lpwd
- get – mget
- put – mput
- open – close – quit – bye
- help – ?

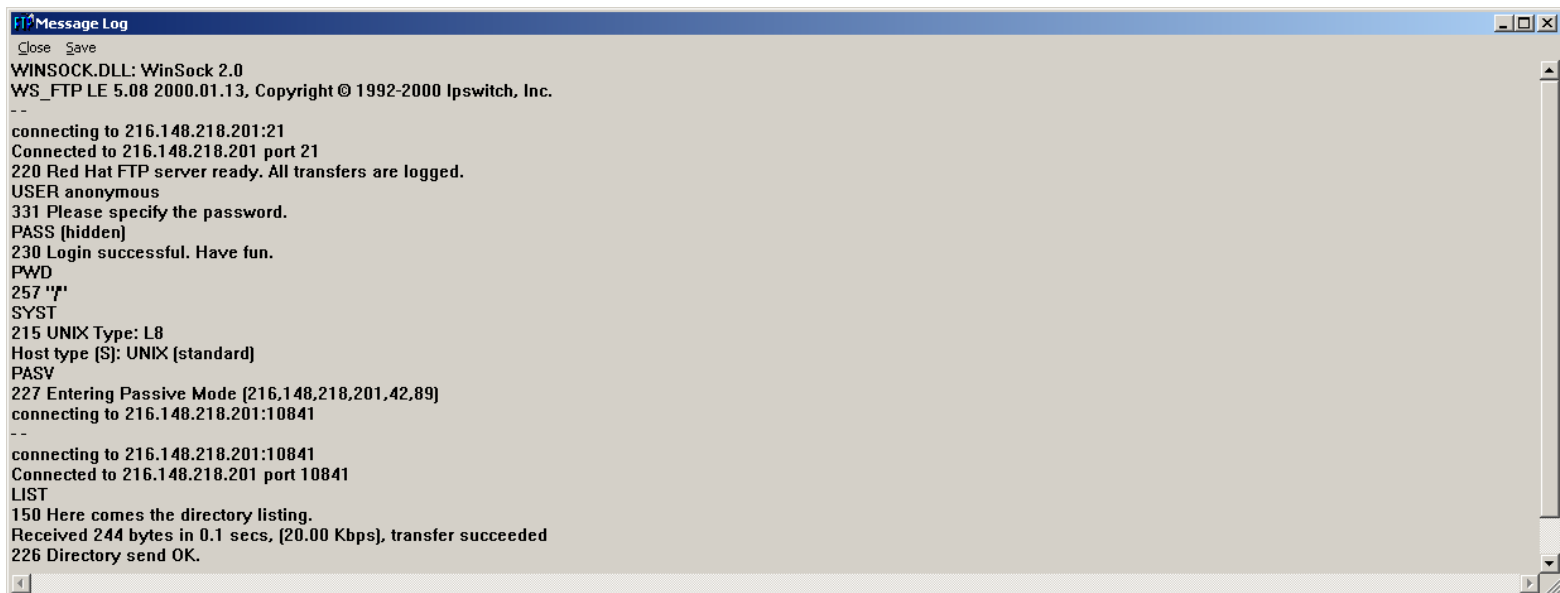
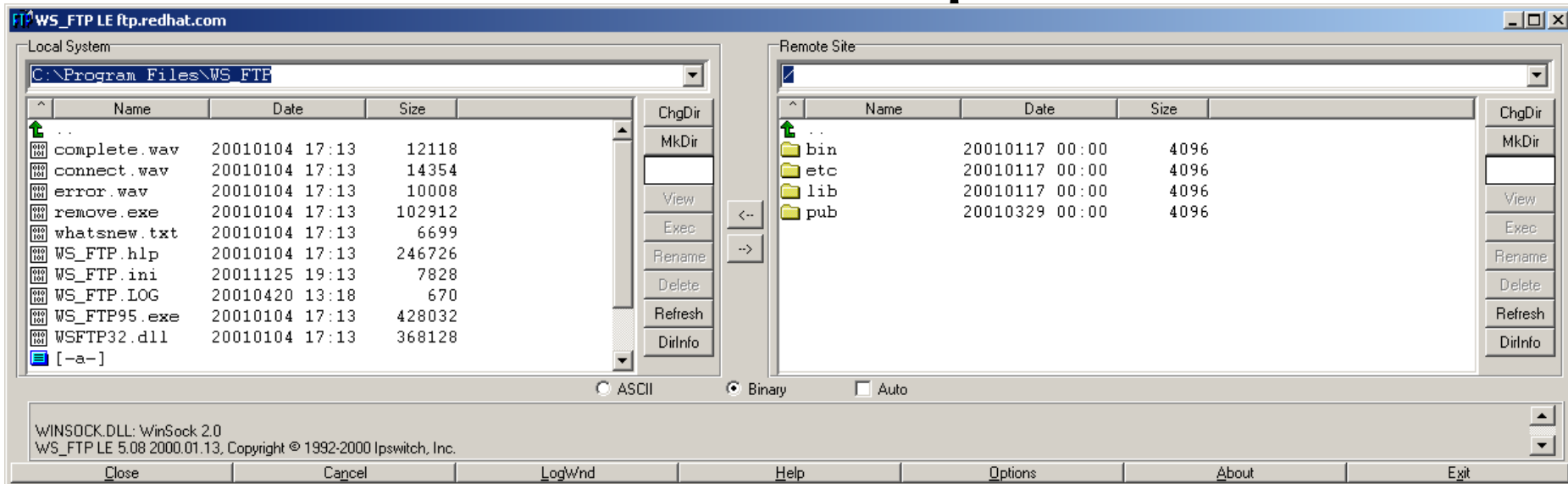
FTP – Exemple 1

```
Command Prompt
C:\>ftp marsu
Connected to marsupilami.brussels.torrentnet.com.
220 marsupilami.brussels.torrentnet.com FTP server (Version wu-2.5.0(1) Tue Sep
21 16:48:12 EDT 1999) ready.
User (marsupilami.brussels.torrentnet.com:(none)): ygancher
331 Password required for ygancher.
Password:
230-
230-
230-*****
230-* Welcome to Marsupilami Brussels's FTP server *
230-* *
230-* Contact: europe@torrentnet.com *
230-*****
230-
230-
230-
230 User ygancher logged in.
ftp> bin
200 Type set to I.
ftp> get ipa.htm
200 PORT command successful.
150 Opening BINARY mode data connection for ipa.htm (3412 bytes).
226 Transfer complete.
ftp: 3412 bytes received in 0.01Seconds 341.20Kbytes/sec.
ftp> help
Commands may be abbreviated. Commands are:

! delete literal prompt send
? debug ls put status
append dir mdelete pwd trace
ascii disconnect mdir quit type
bell get mget quote user
binary glob mkdir recv verbose
bye hash mls remotehelp
cd help mput rename
close lcd open rmdir
ftp> quit
221-You have transferred 3412 bytes in 1 files.
221-Total traffic for this session was 4138 bytes in 1 transfers.
221-Thank you for using the FTP service on marsupilami.brussels.torrentnet.com.
221 Goodbye.

C:\>
```

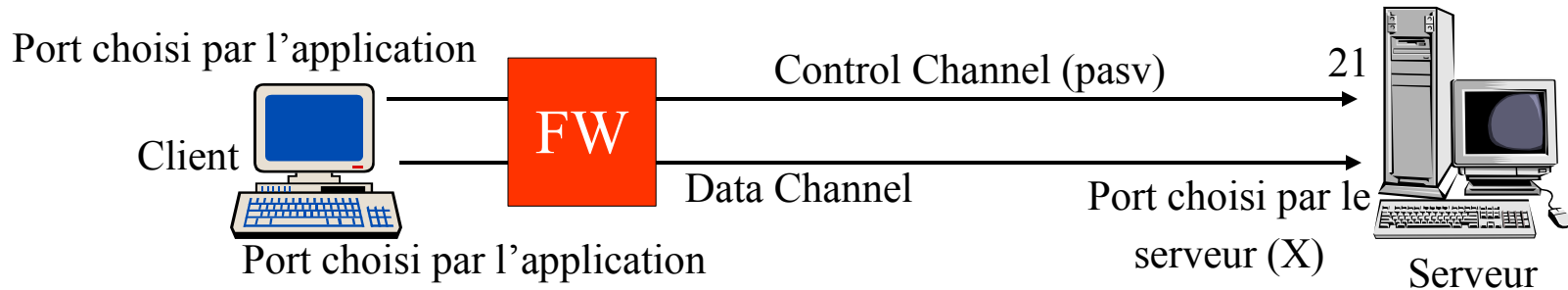
FTP – Example 2



FTP – Mode Passif

- En mode Passif, le client ne spécifie pas de commande 'PORT' qui engendre un session (Data Channel) initiée par le serveur (sécurité).
- A la place, le client envoie un commande 'PASV' qui demande un numéro de port au serveur.
- Le serveur répond par un numéro de port qui pourra être utilisé pour que le client initie la connexion sur le Data Channel. Le serveur écoute donc sur ce nouveau port.
- C'est donc toujours le client qui initie les sessions.

FTP – Passif – Fonctionnement



TFTP

- Trivial File Transfer Protocol
- Utilisé pour transférer un fichier d'une machine à l'autre (cf FTP)
- UDP port 69
- Aucune sécurité
- Simple et facile

TFTP – Détails

- RFC 1350
- Sur port UDP, donc pas de fiabilité
- Développé pour avoir une possibilité de transfert de fichier facile à implémenter !
- Parfois utilisé pour le bootstrap

HTTP

- Hyper Text Transfer Protocol
- Utilisé pour afficher des informations d'une machine sur une autre
- TCP port 80
- Pas de sécurité à la base
- Vu le succès (!!!) il est encore en pleine évolution
- v1.1 standardisée dans RFC 2616

HTTP – Détails 1

- Basé sur deux concepts:
 - URL (Uniform Resource Locator) – Hyperliens
 - Définis dans RFC 1738
 - HTML (Hyper Text Markup Language)
- Sans connexion et sans état (connectionless and stateless)

HTTP – Détails 2

- Différentes méthodes sont utilisées:
 - GET: demande d'un document
 - HEAD: demande de l'entête d'un document (caching)
 - RESPONSE: réponse à une demande (contient généralement de l'HTML)
 - ...

HTTP – URL

- But: avoir un nom unique pour chaque document.
A la base, vient du partage des documents scientifiques
- Structure:
 - `<Protocole>:// < username>: < password>@< nom_de_machine>< :port>/ < répertoire>/ < nom_de_fichier>`
- Protocole: ftp; http; mailto; news; nntp; telnet ...
- Ex:
 - `http://www.redhat.com/`
 - `http://adalbert:tirouflon@purefm.be:1345/adalbert/index.html`
 - `ftp://ftp.redhat.com`
 - ...

HTTP – HTML

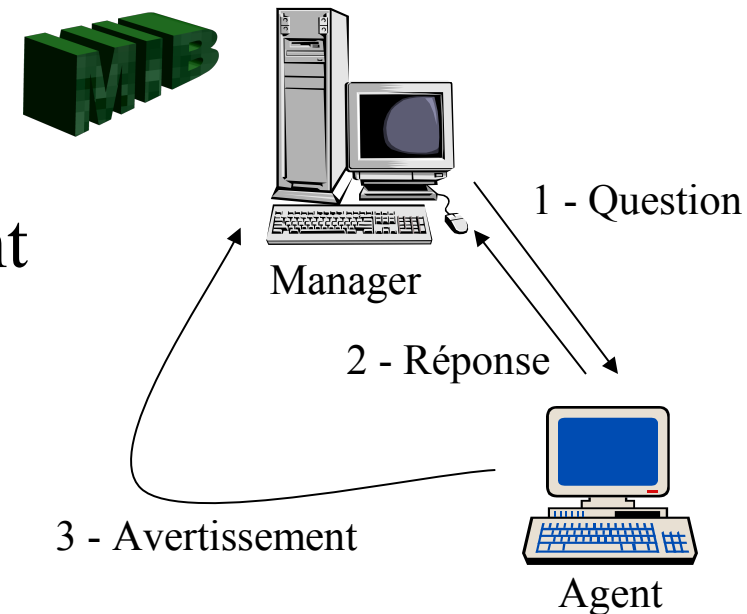
- Langage structuré, sous-produit de SGML, permettant l'indépendance entre le formatage du document et la machine sur laquelle il est lu !
- Voir autre cours !
- Basé sur la notion de requête – réponse (cf client – serveur)...

SNMP

- Simple Network Management Protocol
- Utilisé pour gérer les machines (routeurs, switches, repeaters, PCs, imprimantes, machines à café...) d'un réseau
- UDP port 161
- v1 (pas de sécurité), v2 (peu de sécurité), v3 (sécurisé mais quasiment pas implémenté)

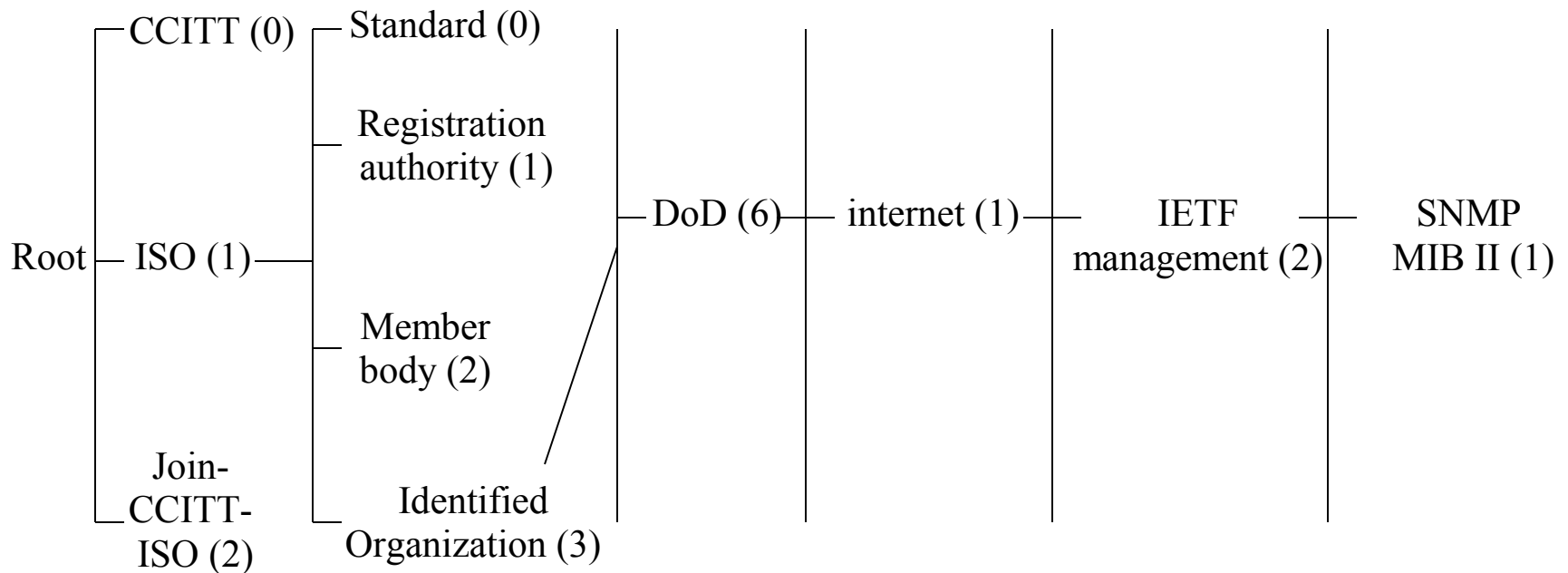
SNMP – Principes

- Basé sur la notion de MIB (publiques – privées) qui donne un ID aux objets
- MIB écrites en ASN.1
- Deux possibilités:
 - Le manager interroge l'agent
 - L'agent avertit le manager
- Possibilité de proxy



SNMP – L'arbre des MIBs

Généralités



Notation: 1.3.6.1.2.1.

SNMP – L'arbre des MIBs

Suite

- 1.3.6.1.2.1.1: System —————
- 1.3.6.1.2.1.2: Interfaces
- 1.3.6.1.2.1.3: at
- 1.3.6.1.2.1.4: IP
- 1.3.6.1.2.1.5: ICMP
- 1.3.6.1.2.1.6: TCP
- 1.3.6.1.2.1.7: UDP
- 1.3.6.1.2.1.11:SNMP
- 1.3.6.1.2.1.14: OSPF v2
- 1.3.6.1.2.1.15: BGP v4
- ...
- 1.3.6.1.2.1.1.1: System Description
- 1.3.6.1.2.1.1.2: System Object ID
- 1.3.6.1.2.1.1.3: System Up Time
- 1.3.6.1.2.1.1.4: System Contact
- 1.3.6.1.2.1.1.5: System Name
- 1.3.6.1.2.1.1.6: System Location
- 1.3.6.1.2.1.1.7: System Services

SNMP v1 – Les opérations

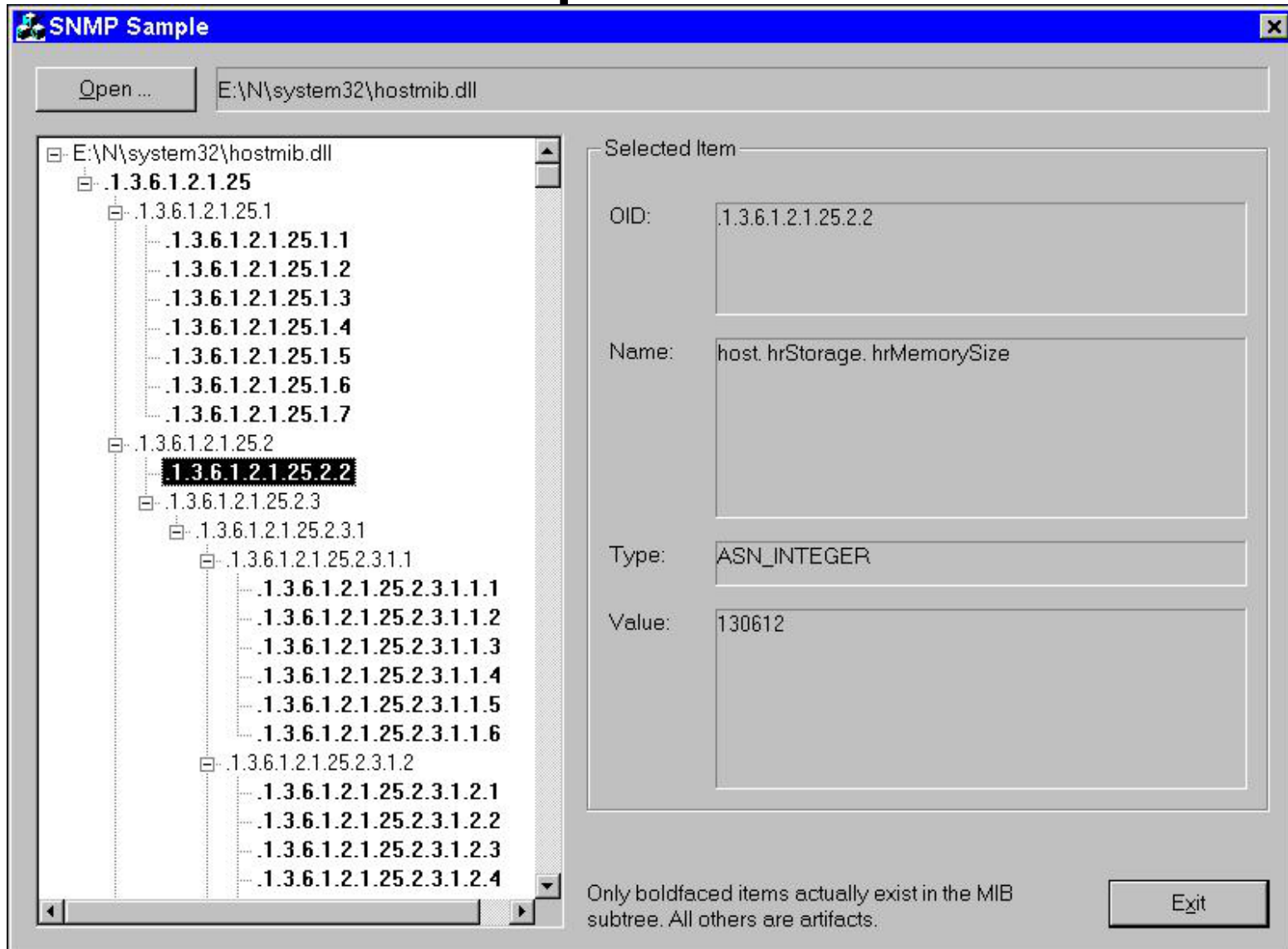
- Get
- Get next
- Get response
- Set
- Trap

SNMP v2c – Les opérations

- Get*
- Get next*
- Get bulk
- Response
- Set*
- Inform
- Trap v2

*: Existant en SNMP v1

SNMP – Exemple de MIB Browser



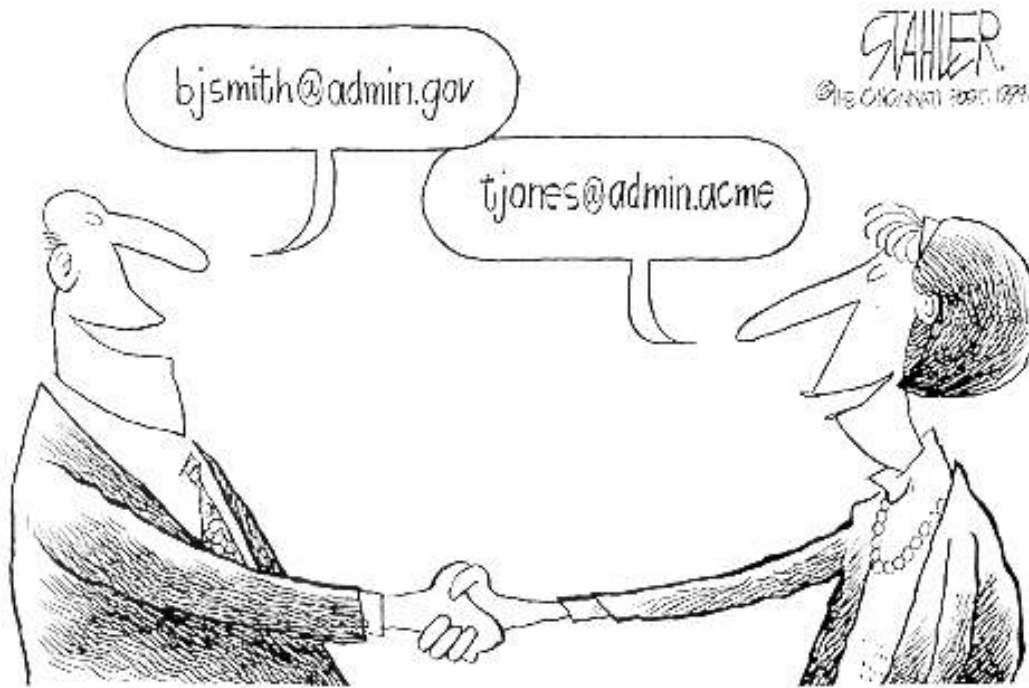
IPFIX

- IP Flow Information Export
- Première version début 2004
- Similaire à SNMP, mais avec la notion de flux, basé sur
 - Port source et dest (niveau 4)
 - IP source et dest (niveau 3)
 - Protocol Type (niveau 3)
 - TOS (niveau 2)
 - Incoming interface (niveau 1)
- Il faut que les devices et la console de management soient 'IPFIX compliant'...

Mail – SMTP

- Simple Mail Transfer Protocol
- Utilisé pour envoyer un mail d'un serveur mail à l'autre
- TCP port 25
- Pas de sécurité
- RFC 821
- Extended SMTP (ESMTP), RFC 1869

Mail – Très populaire



T'as pas reçu mon Email ?...

Mail – Adresse et Spooling

- Adresse: username@domainname. Mélange entre
 - Un nom d'utilisateur
 - Un nom de domaine
- Alias, liste de distribution
- Les machines peuvent ne pas être connectées au même moment, donc spooling (gestion de queues et essais à intervalles réguliers)

Mail – MTA – MUA

- MTA
 - Message Transfer Agent
 - Utilisé pour recevoir les mails et les fournir aux utilisateurs
 - Responsable du routage des messages.
 - Partie serveur (ex: Unix sendmail et MS Exchange Server)
 - Implémente SMTP
- MUA: Message User Agent.
 - Message User Agent
 - Utilisé pour recevoir, présenter, préparer et envoyer les mails
 - Partie client (ex: Thunderbird, Eudora, Outlook)
 - Implémente POP ou / et IMAP

Mail – Structure d'un message

- Enveloppe
 - Utilisée par les MTAs pour la livraison
 - Emetteur, Récepteur(s)
- En-tête
 - Utilisé par les MUAs
 - Message ID, Date / Heure, Mailer System...
- Corps
 - Contenu du message

Commandes SMTP (référence)

- HELO <domain>
- MAIL FROM:<reverse-path >
- RCPT TO:<forward-path>
- DATA
- QUIT
- RSET
- VRFY <string>
- SEND FROM:<reverse-path>
- SOML FROM:
 <reverse-path>
- SAML FROM:
 <reverse-path>
- EXPN <string>
- HELP [<string>]
- NOOP
- TURN

Mail – Les trois paradigmes du client

- Définis dans la RFC 1733
- Off-line: les messages sont téléchargés sur la machine du client puis effacé du serveur. Le client se connecte régulièrement au serveur pour voir si de nouveaux messages sont arrivés. C'est donc un modèle de type 'store and forward'. La machine client est souvent unique, même si plusieurs mailbox (pour différents clients) coexistent sur le serveur.
- On-line: les messages sont gérés directement sur le serveur. Les messages ne sont pas stockés sur la machine du client. Cela permet une indépendance du lecteur par rapport à la machine: on peut lire sa mailbox de n'importe où ! Plus évolué que le modèle off-line.
- Déconnecté: hybride des deux précédents, le mode déconnecté downloade les messages sur la machine client, les manipule localement puis resynchronise lors d'une nouvelle connexion. Attention aux problèmes lors d'accès depuis plusieurs clients.

Mail – POP

- Post Office Protocol
- Sert à aller chercher les messages sur le serveur
- N'implémente que le paradigme « off-line »
- Permet aussi d'utiliser un mode « pseudo on-line », où les messages sont laissés sur le serveur
- V3
- RFC 1725
- Pas de sécurité
- TCP port 110
- Le plus ancien et le plus utilisé, et de loin !

Mail – DMSP

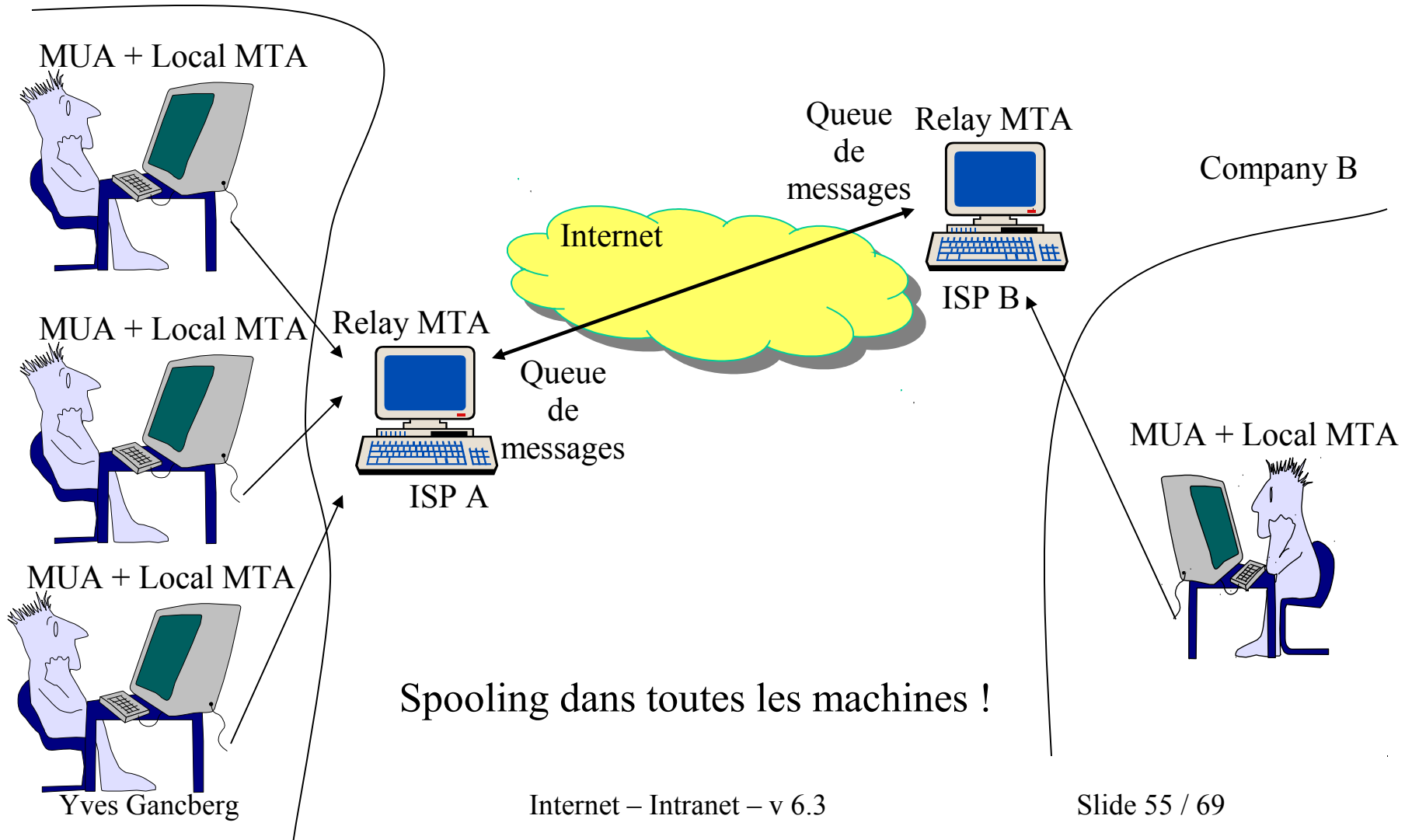
- Distributed Mail System Protocol
- Implémente le paradigme « déconnecté »
- Une seule application: PCMAIL.
- Quasiment pas utilisé

Mail – IMAP

- Internet Message Access Protocol
- Implémente les trois paradigmes (« on-line », « off-line » et « déconnecté ») !!!
- Sert à consulter les messages sur le serveur
- Plus complet que POP3 (Superset de POP et de DMSP) puisqu'il implémente les trois paradigmes
- V4
- RFC 2060
- TPC port 143
- Offre plus de possibilités au niveau des flags associés à un message (lu, supprimé, répondu...)
- Permet également de ne demander qu'une partie du message (en-tête, corps, correspondance à certains critères...)
- Possibilité de manipulation de plusieurs mailbox en même temps, ou par plusieurs utilisateurs en même temps.

Mail – Exemple pratique

Company A



Mail – X400

- Décrit par le CCITT
- Complexe (!)
- Agonisant...

Mail – Politique

- Généralement, les ISP refusent de faire du ‘SMTP relay’ vers un autre ISP.
- Pourquoi ? Contre les spammeurs !
- Donc, le serveur POP et le serveur SMTP peuvent être différents.
- En changeant d’ISP, on risque de devoir changer de serveur SMTP.

Mail – Exemple de mail forgé

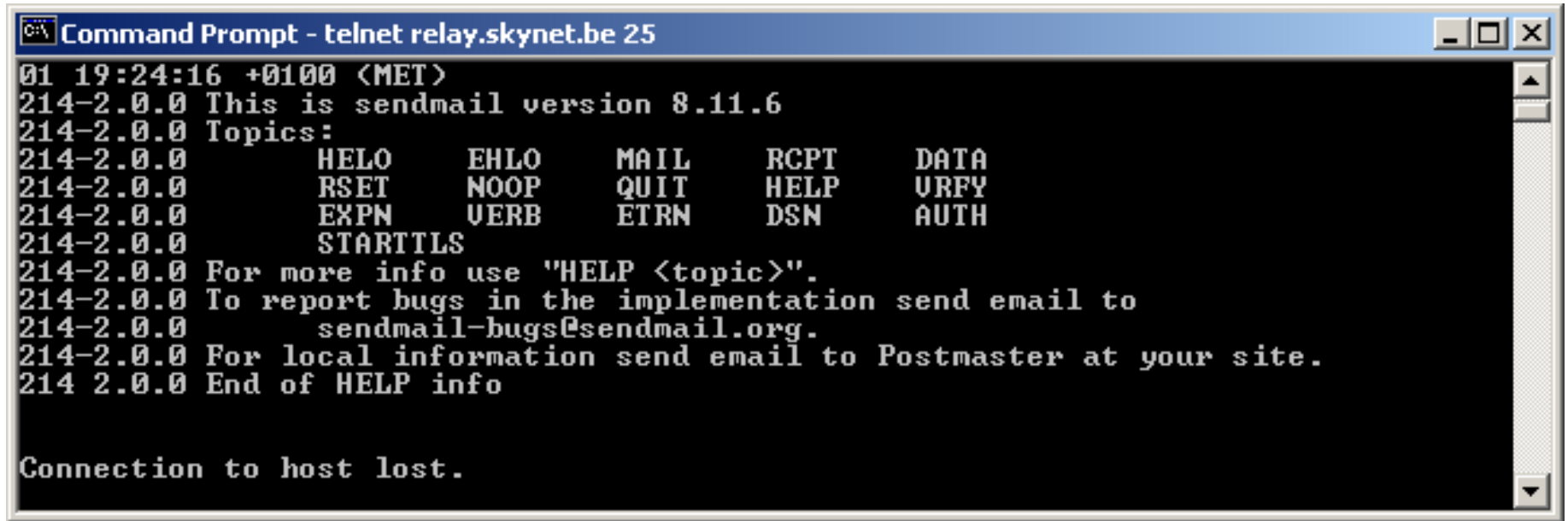
En tête SMTP

- Delivered-To: XXX@gmail.com
- Received: by 10.229.191.17 with SMTP id dk17cs40455qcb;
 - Thu, 9 Dec 2010 09:51:36 -0800 (PST)
- Received: by 10.224.73.195 with SMTP id r3mr3791920qaj.337.1291917095675;
 - Thu, 09 Dec 2010 09:51:35 -0800 (PST)
- Received-SPF: softfail (**google.com: best guess record for domain of transitioning YYY@hotmail.com does not designate 65.55.90.24 as permitted sender**) client-ip=65.55.90.24;
- Received: by 10.241.83.157 with POP3 id 29mf278099qyk.36;
 - Thu, 09 Dec 2010 09:51:35 -0800 (PST)
- X-Gmail-Fetch-Info: XXX@hotmail.com 1 pop3.live.com 995 XXX@hotmail.com

Mail – Connection SMTP manuelle

- Telnet smtp.mycompany.com 25
- Commandes:
 - Mail from: <adresse source>
 - Rcpt to: <adresse destination>
 - Data <contenu du message>
 - Quit
 - ...

Mail – Connection SMTP manuelle – Exemple



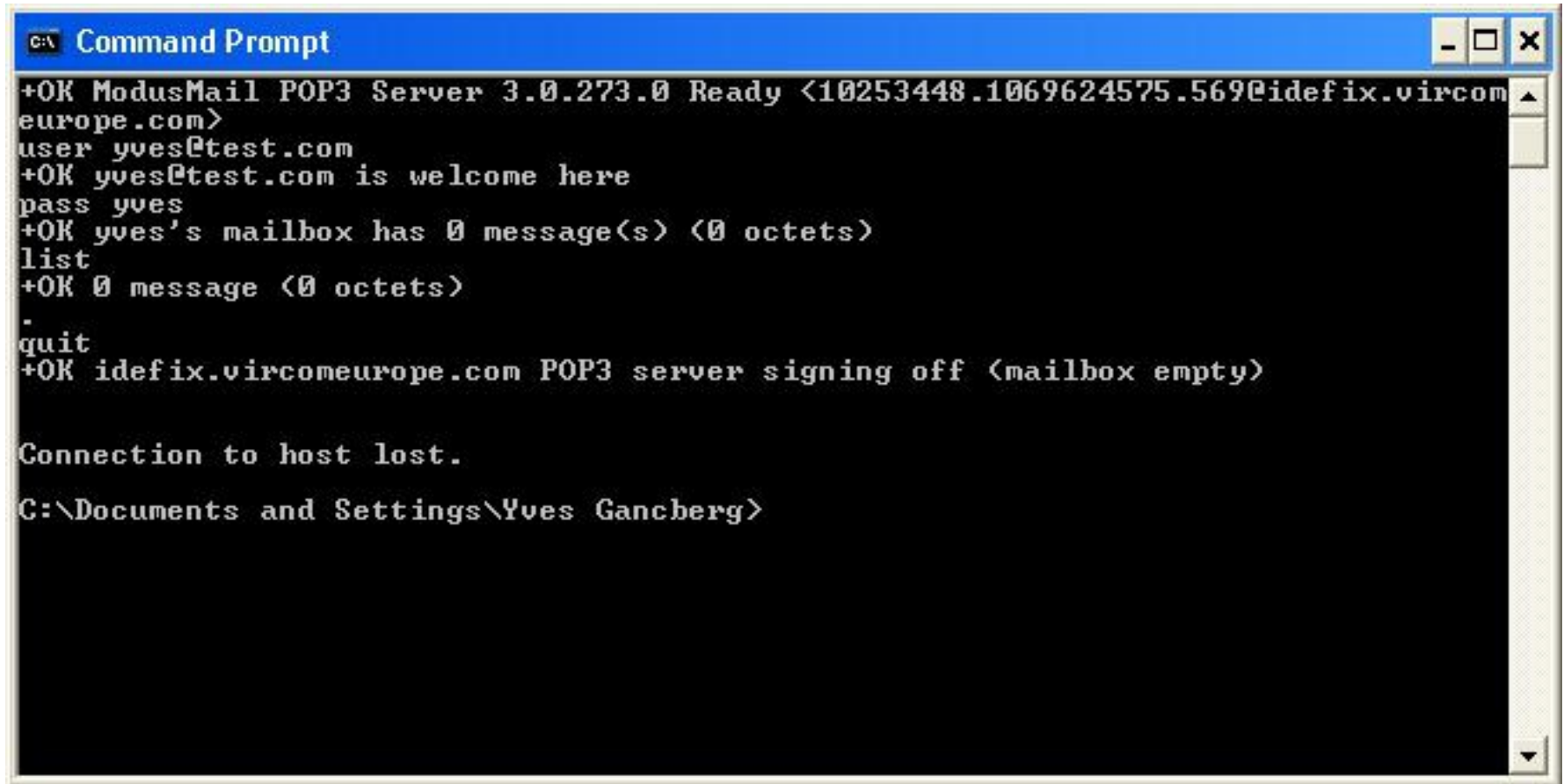
```
C:\> telnet relay.skynet.be 25
01 19:24:16 +0100 <MET>
214-2.0.0 This is sendmail version 8.11.6
214-2.0.0 Topics:
214-2.0.0      HELO      EHLO      MAIL      RCPT      DATA
214-2.0.0      RSET      NOOP      QUIT      HELP      URFY
214-2.0.0      EXPN      VERB      ETRN      DSN       AUTH
214-2.0.0      STARTTLS
214-2.0.0 For more info use "HELP <topic>".
214-2.0.0 To report bugs in the implementation send email to
214-2.0.0      sendmail-bugs@sendmail.org.
214-2.0.0 For local information send email to Postmaster at your site.
214 2.0.0 End of HELP info

Connection to host lost.
```

Mail – Connection POP manuelle

- Telnet pop.mycompany.com 110
- Commandes:
 - User <username>
 - Pass <password> (!)
 - List
 - Quit
 - ...

Mail – Connection POP manuelle – Exemple



```
C:\ Command Prompt
+OK ModusMail POP3 Server 3.0.273.0 Ready <10253448.1069624575.569@idefix.vircomeurope.com>
user yves@test.com
+OK yves@test.com is welcome here
pass yves
+OK yves's mailbox has 0 message(s) <0 octets>
list
+OK 0 message <0 octets>
.
quit
+OK idefix.vircomeurope.com POP3 server signing off <mailbox empty>

Connection to host lost.
C:\Documents and Settings\Yves Gancberg>
```

NTP

- Network Time Protocol
- Sert à synchroniser les horloges de différentes machines à travers le réseau.
- UDP port 123.
- V3 standardisé dans la RFC 1305.
- V4 (transformé en SNTP) dans la RFC 2030.
- 10 à 20 millions de clients...
- Précis à quelques millisecondes près...

Peer To Peer

- Nouveauté
- Partage en ligne de musique, de films, de fichiers en général
- Pas de port standard
- Engouement des jeunes (c'est vous ! :-))
- Problèmes légaux (le programme ou le site est légal, mais pas l'utilisation qu'on en fait)
- Exemples: (Napster), Kazaa, Emule - Edonkey, BitTorrent...

Peer to Peer – Chiffres

- Enorme quantité de trafic généré (2008: 40 Milliards de morceaux de musique téléchargés illégalement...) !
- Le Soir : jeudi 26 août 2010 : un internaute télécharge 2.6 Téraoctets en 1 mois !!!
- En 2011, un créateur (musicien) qui fait un CD vendu 20 € dans le commerce touche entre 2 et 5 € (selon le type de contrat). Soit entre 20 et 30 % du prix hors commission revendeur ($20 - 13 - 1.3 \times 2$). Sur un site de partage (genre MySpace, iTunes etc...), ce % monte à 80 – 90 %.
- On comprends pourquoi les Majors n'aiment pas cela ! Cela fait autant de milliards de \$ de revenus en moins !

Peer to Peer – L'aspect légal

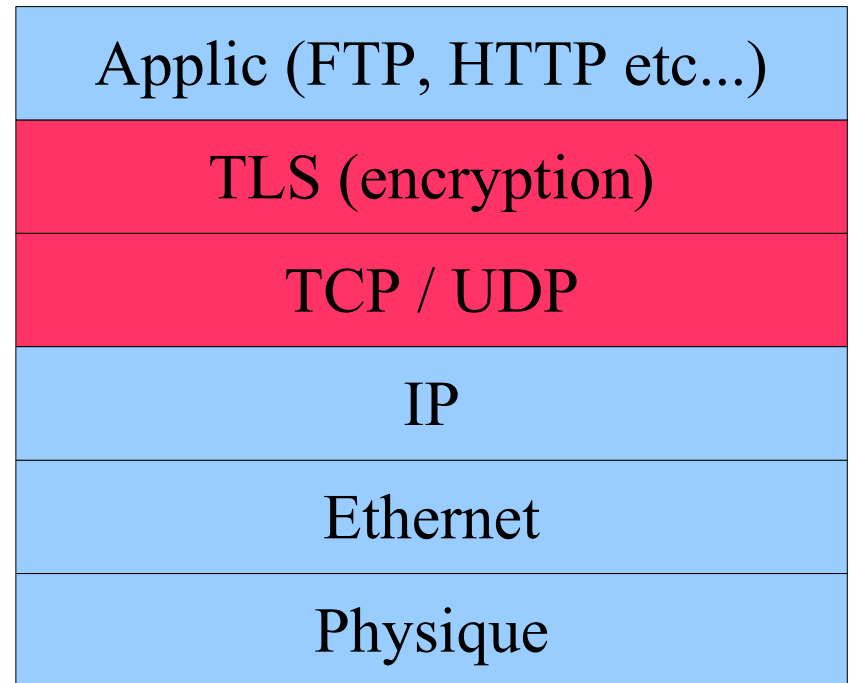
- La Belgique essaie (avec l'aide des ISP) de bloquer l'accès à certains sites de diffusion de Torrents
- Succès grandissant
- Existe-t-il des parades à la parade ?
- Le réseau TOR ?

Sécuriser des applications non sécurisées...

- Il y a un ensemble d'applications (FTP, HTTP, RTP (voir plus loin), POP etc...) qui ne sont pas sécurisées.
- Une manière de les sécuriser est de rajouter une couche d'encryption avant de transmettre les données à la couche transport
- Cette couche, appelée SSL ou TLS, a été initialement développée par Netscape

TLS – Transport Layer Security

- Initialement développé par Netscape
- V2, V3, puis... faillite
- Ensuite repris sous le nom TLS, utilisé aujourd'hui
- Le nouveau protocole (FTPS, HTTPS etc...) reçoit un nouveau port de l'IANA
- TLS : niveau 5 (OSI) : Session



Commotion – Réseau Citoyen

- Sascha Meinrath est à la fois fataliste et optimiste : « Que ce soit aux Etats-Unis, au Moyen-Orient ou ailleurs, qui va mettre en place ces réseaux alternatifs ? Pas des vieux, on le sait. Ce sont les ados qui vont s'en emparer. Ils s'en serviront pour contester l'ordre établi et aussi pour partager leur musique et leurs films. Ce sera peut-être négatif pour les détenteurs de droits, mais le bilan global sera très positif ».