

NFA016 : Les protocoles

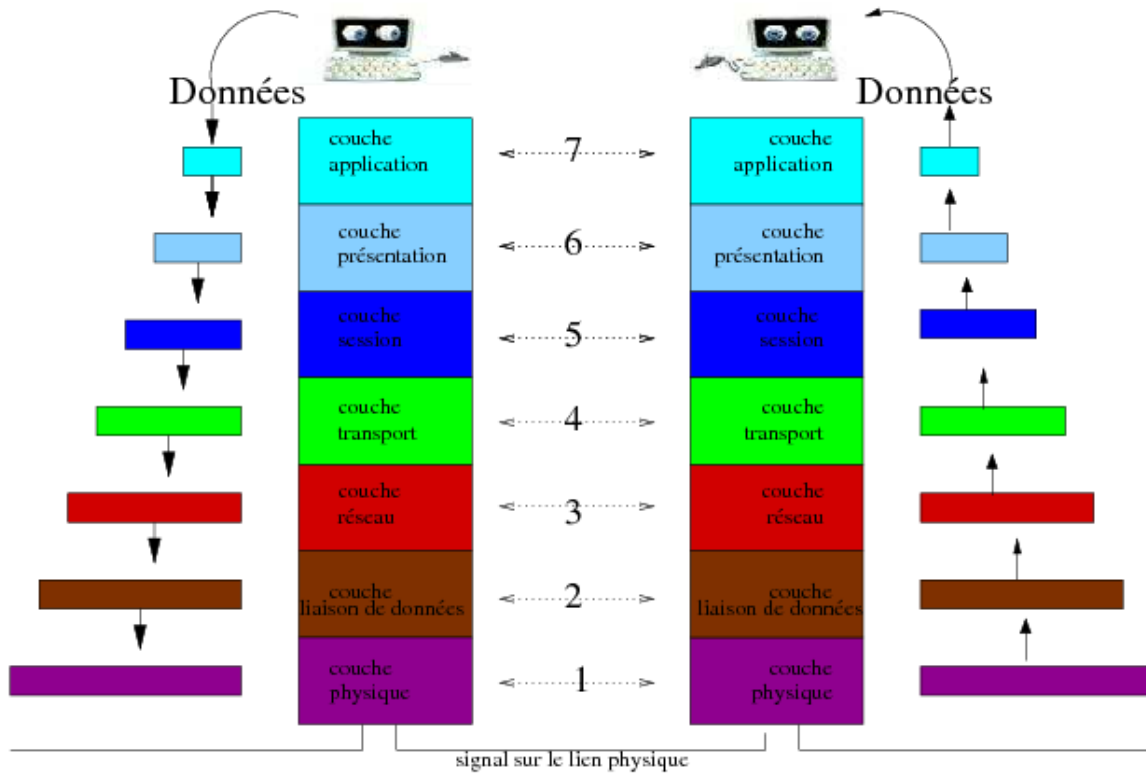
O. Pons, S. Rosmorduc

Conservatoire National des Arts & Métiers

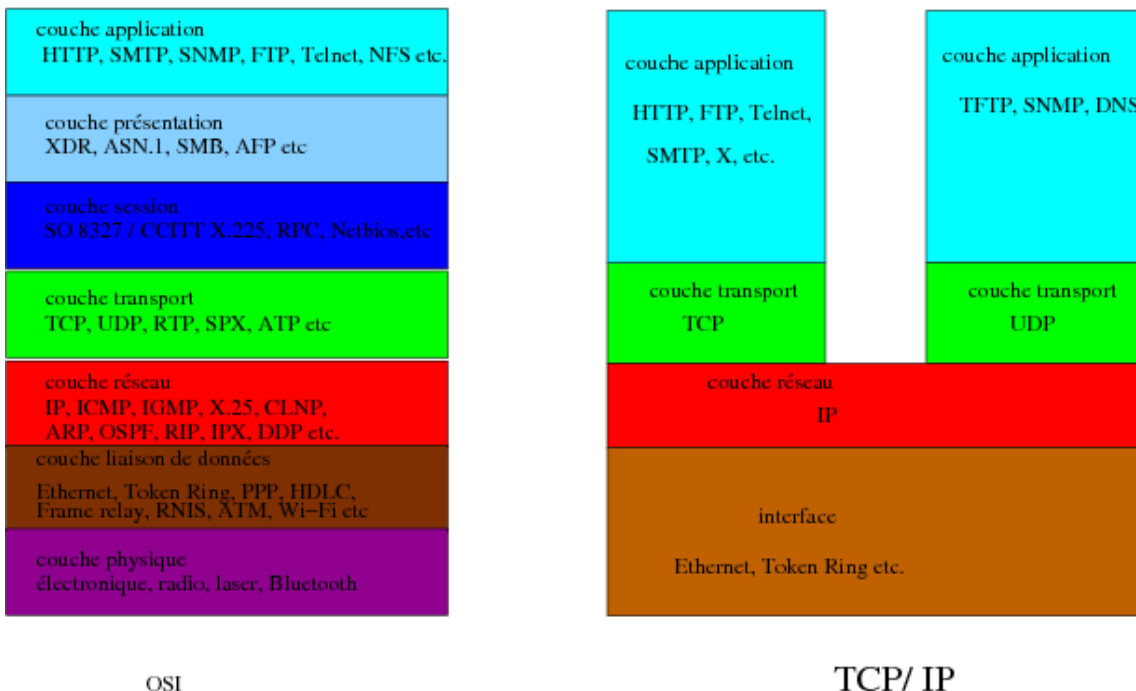
Notion de protocole

- Un réseau: ensemble de machines reliées entre elles.
- Communiquer:
 - Parler "la même langue".
 - Protocole de communication.
 - Les protocoles dépendent du niveau où l'on se place: Abstraction
- Organisation en couches :
 - Modèle OSI
 - TCP/IP

Modèle OSI



TCP/IP



Identification d'une machine

- Numéro IP : *193.54.195.244*
- Nom de machine : *www.ensiie.fr*
- DNS, résolution de nom

Une tentative de connexion:

```
[olivier@mamaison tmp]$ telnet www.ensiie.fr
Trying 193.54.195.244...
telnet:connect to address 193.54.195.244: Connection refused
telnet:Unable to connect to remote host: Connection refused
[olivier@mamaison tmp]$
```

Identification d'un service : Port

- Numéro de port : identifie un service
- Si le serveur **écoute sur un port** la connexion est possible.
- Si un **daemon** sshd tourne sur le port 22 sur **www.ensiie.fr**

```
[olivier@mamaison tmp]$ telnet www.ensiie.fr 22
Trying 193.54.195.244...
Connected to www.ensiie.fr (193.54.195.244).
Escape character is '^]'.
SSH-1.99-OpenSSH_3.8.1p1 Debian-8.sarge.4
....
```

- Si un serveur http tourne sur le port 80 sur **www.ensiie.fr**

```
[olivier@mamaison tmp]$ telnet www.ensiie.fr 80
Trying 193.54.195.244...
Connected to www.ensiie.fr (193.54.195.244).
Escape character is '^]'.
....
```

Protocol et URL (Uniform Ressource Locator)

- Protocol : "language" utilisé pour communiquer **une fois connecté**
- URL : `<protocol>:<partie dépendante du protocole>` Forme générale:

```
http    //<machine>:<port>/chemin?value
ftp     //<utilisateur>:<motdepasse>@<machine>:<port>/reps/fichier
file    //chemin
mailto  adresse-mail
telnet  //<utilisateur>:<motdepasse>@<machine>:<port>
```

Exemples:

```
http://www.ensiie.fr/~pons/COURS/cgi-bin/toto.cgi?moi=olivier
telnet://pons@lunix121.iie.cnam.fr:22
mailto:pourlesnews@yahoo.fr
ftp://ftp.ens-cachan.fr/mirror/Mandrakelinux/
file://home/pons/tmp.html
```

- RCF (Request for comment), documents de description des protocoles

Dialoguer "à la main" avec un serveur

1. se connecter sur le port correspondant au service
2. dialoguer en respectant le protocole

Exemple: L'envoi de mails, SMTP

```
[inviteUnix~]$telnet smtp.free.fr 25
Trying 212.27.48.4...
Connected to smtp.free.fr.
Escape character is '^]'.
220 smtp1-g19.free.fr ESMTP Postfix
EHLO olivier
250-smtp1-g19.free.fr
...
250 8BITMIME
MAIL FROM: pons@cnam.fr
250 Ok
RCPT TO: pourlesnews@yahoo.fr
250 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
test pour le cours protocols...
C beau.
.
250 Ok: queued as A341D3480
QUIT
221 Bye
Connection closed by foreign host.
[inviteUnix~]$
```

Exemple: L'envoi de mails, SMTP 2

Fixer les champs **from** et **to** :

```
HELO moi
MAIL FROM: pons@cnam.fr
```

```
RCPT TO: pons@cnam.fr
DATA
From: segolene@ps.fr
To: jacque@elyse.fr
Subject: un test
```

```
blabla
```

```
.
QUIT
```

Exemple: L'envoi de mails, SMTP 3

- envoyer des images, des textes avec accent, des caractères plus compliqués ...
- Type MIME (Multipurpose Internet Mail Extensions)

```
EHLO moi
MAIL FROM:pons@cnam.fr
RCPT TO:pons@cnam.fr
DATA
From: candidats@cnam.fr
To: leprochain@elyse.fr
Subject:"bientot moi ..."
MIME-Version: 1.0
Content-Type: image/jpg; name="arrow.gif"
Content-Transfer-Encoding: base64
Content-Disposition: attachment;
  filename="arrow.gif"

R0lGODlhBwAEAIAAAAAAAAAP///yH5BAEAAAEALAAAAAAHAAQAAAIiHa+BGWON
WSgAOw==
====

.
QUIT
```

Exemple: La reception de mails, POP

```
pons@pc-olive:~ telnet pop.mail.yahoo.fr 110
Trying 217.12.10.100...
Connected to pop1.mail.vip.ukl.yahoo.com.
Escape character is '^]'.
+OK hello from popgate(2.35.8)
USER pourlesnews
+OK password required.
PASS monmotdepasse
+OK maildrop ready, 83 messages (498477 octets) (80067
1073741824)
LIST
+OK 83 messages (498477 octets)
1 16115
2 943
3 1189
4 4021
5 2007
6 4922
7 762
...
```

Exemple: La réception de mails, POP 2

```
...
TOP 7 10
+OK 762 octets
X-Apparently-To: pourlesnews@yahoo.fr via 212.12.10.165; Mon, 30 Oct 2006 03:40:
44 -0800
X-Originating-IP: [212.27.42.27]
Authentication-Results: mta517.mail.mud.yahoo.com from=cnam.fr; domainkeys=neutral (no sig)
Received: from 212.27.42.27 (EHLO smtp1-g19.free.fr) (212.27.42.27)
by mta517.mail.mud.yahoo.com with SMTP; Mon, 30 Oct 2006 03:40:44 -0800
Received: from olivier (nor75-7-81-57-85-121.fbx.proxad.net [81.57.85.121])
by smtp1-g19.free.fr (Postfix) with ESMTP id A341D3480
for ; Mon, 30 Oct 2006 12:39:13 +0100 (CET)
Message-Id: <20061030113913.A341D3480@smtp1-g19.free.fr>
Date: Mon, 30 Oct 2006 12:39:13 +0100 (CET)
From: pons@cnam.fr
To: undisclosed-recipients;;

test pour le cours protocol...
C beau

.
QUIT
+OK server signing off. Connection closed by foreign host.
```

Exemple: le web, HTTP

```
pons@pc-olive:~$ telnet rdp-pentium.iie.cnam.fr 80
Trying 193.54.225.178...
Connected to rdp-pentium.ensiie.fr.
Escape character is '^]'
GET /users/pons/www/exCours.html

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 3.2//EN">
<html>
<head>
  <title>Exemple pour cours http</title>
</head>

<body>
  <p>mon exemple</p>C beau
</body>
</html>
Connection closed by foreign host.
```

HTTP

Propose différentes **méthodes** qui dépendent des versions

- HTTP original (0.9) : **GET** et connexion fermée
- HTTP 1.0 : ajout de **HEAD** et **POST** + ...
- HTTP 1.1 : ajout de **OPTIONS**, **CONNECT** et **TRACE**

Exemple: HTTP

```
pons@pc-olive:~$ telnet rdp-pentium.iie.cnam.fr 80
Trying 193.54.225.178...
Connected to rdp-pentium.ensiie.fr.
Escape character is '^]'.
HEAD /exCours.html HTTP/1.0

HTTP/1.1 200 OK
Date: Mon, 30 Oct 2006 04:19:05 GMT
Server: Apache-AdvancedExtranetServer/2.0.47 (Mandrake Linux/1.9.9lmdk) mod_perl/1.99_08 Perl/v5.8
Last-Modified: Mon, 30 Oct 2006 02:32:44 GMT
ETag: "4b099-50-15a8ab00"
Accept-Ranges: bytes
Content-Length: 80
Connection: close
Content-Type: text/html; charset=ISO-8859-1

Connection closed by foreign host.
pons@pc-olive:~$
```

Exemple: HTTP

```
[olivier@mamaison]$ telnet www.ensiie.fr 80
Trying 193.54.195.244...
Connected to www.ensiie.fr (193.54.195.244).
Escape character is '^]'.
POST /cgi-bin/lecture.cgi?moi=oliver&elle=marianne HTTP/1.0
Host: localhost
Content-Type: application/x-www-form-urlencoded
Content-Length: 30

vous=Les+eleves&eux=Les+autres
HTTP/1.1 200 OK
Date: Sun, 19 Nov 2006 00:58:27 GMT
Server: Apache/1.3.33 (Debian GNU/Linux) mod_ssl/2.8.22 OpenSSL/0.9.7e mod_perl/1.29 PHP/4.3.10-16
Connection: close
Content-Type: text/html; charset=iso-8859-1

elle === marianne

vous === Les eleves

eux === Les autres

moi === oliver

Connection closed by foreign host.
```

Exemple: HTTP

```
pons@pc-olive:~$ telnet rdp-pentium.iie.cnam.fr 80
Trying 193.54.225.178...
Connected to rdp-pentium.ensiie.fr.
Escape character is '^]'.
OPTIONS /exCours.html HTTP/1.1
Host: rdp-pentium.iie.cnam.fr

HTTP/1.1 200 OK
```

```
Date: Mon, 30 Oct 2006 04:25:33 GMT
Server: Apache-AdvancedExtranetServer/2.0.47 (Mandrake Linux/1.9.91mdk) mod_perl/1.99_08 Perl/v5.8
Allow: GET,HEAD,POST,OPTIONS,TRACE
Content-Length: 0
Content-Type: text/html; charset=ISO-8859-1
X-Pad: avoid browser bug
```

Connection closed by foreign host.

{Codes de retour}

- 20X : OK
- 40X,50X : Problèmes

CGI

Programme qui écrit sur sa sortie standard, le corps de ce qui va être envoyé au client par le serveur

Content-type: text/html

```
<html>
  <head>
    <title>page HTML produite par mon script CGI </title>
  </head>
  <body>
    <h1>Encore un exemple</h1>
    que pensez vous de <strong>ça </strong>
  </body>
</html>
}
```

CGI

- En bash

```
#!/bin/bash
echo "Content-type: text/html"
echo
echo "En shell"
```

- en perl

```
#!/usr/bin/perl
printf("Content-type: text/html\n\n");

printf("En Perl \n");
```

CGI

- en C

```
/* exempleCGI.c */
#include

int main(void){
  printf("Content-type: text/html\n\n");
  printf("En langage C \n");
  return 0;
}
```


}

CGI

- Java

```
/*
pour compiler :
javac exampleEnJava.java
gcj -g --main=ExampleEnJava -o ExempleEnJava.cgi ExampleEnJava.class
*/
class ExampleEnJava {
    public static void main (String[] args){
        System.out.println("Content-type: text/html\n\n");
        System.out.println("En java \n");
    }
}
```

- en Caml

```
/* compiler avec ocamlc */
let rec fac n = if (n<1) then 1 else n*fac(n-1);;

Printf.printf "Content-type: text/html\n\n";;
Printf.printf "
```

un peu de calcul

```
\n";;
Printf.printf "factoriel 10 vaut %d\n" (fac 10);;
```

Le passage des paramètres

- Variables d'environnement

- **REQUEST_METHOD** : la méthode utilisée
- **QUERY_STRING** : les données dans l'url
- **CONTENT_LENGTH** : la longueur des autres données
- **CONTENT_TYPE** ,**HTTP_USER_AGENT**,...

Exemple de traitement cgi

```
#!/usr/bin/perl

###fonction d'analyse des arguments renvoi
###une liste d'association nom/valeur
sub Parse_Form {

#recuperer les argument en fonction de la methode
#####
if($ENV{'REQUEST_METHOD'} eq 'GET') {
    #c'est GET on va decouper la ligne de commande
    #les valeurs sont separee par &
    #on les mets dans un tableau
    @paires = split (/&/, $ENV{'QUERY_STRING'});
}

#sinon ca doit etre POST mais verifions
elseif($ENV{'REQUEST_METHOD'} eq 'POST') {
```

```

#c'est post on va lire sur l'entree standard
# content length permet de detecter la fin
read(STDIN, $buffer, $ENV{'CONTENT_LENGTH'});

@paires = split (/&/, $buffer);
#si il y en a aussi dans l'url
if ($ENV{'QUERY_STRING'}){
    @getpaires = split (/&/,$ENV{'QUERY_STRING'});
    push(@paires,@getpaires);
}
}
#si on est la c'est une autre methode
else{
    print("Content-type: text/html\n\n");
    print $ENV{'REQUEST_METHOD'};
    print "n'est pas connue utiliser GET ou POST";
}

##traitement des paires
#####
#pour chaque paire

foreach $paire (@paires) {

    #on separe le nom et la valeur
    ($nom,$val) = split (/=/,$paire);
    #remplace le signe plus par un blanc
    $nom =~ tr /+ / /;

    #Berk conversion de l'hexa a la representation usuel
    $nom =~ s/%([a-fA-F0-9][a-fA-F0-9])/pack("C",hex($1))/eg;

    #remplace le signe plus par un blanc
    $val =~ tr /+ / /;
    #Berk conversion de l'hexa a la representation usuel
    $val =~ s/%([a-fA-F0-9][a-fA-F0-9])/pack("C",hex($1))/eg;

    #elimine les inclusion possible venant de serveur ( SSI)
    $val=~s///g;

    ##formadata est le nom de la liste d'association (hash)
    ## que l'on construit
    ##si le mon a deja une value (ex mutliple value)
    ##on ajoute la nouvelle

    if($formdata{$nom}){
        $formdata{$nom} .= ", $val";
    }
    ##sinon on met juste la valeur
    else {
        $formdata{$nom} = $val;
    }
}

}%formdata;
}

##pour tester

print("Content-type: text/html\n\n");

my %donnees = &Parse_Form();

```

```
@clefs = keys(%donnees);
foreach $clef (@clefs){
    $val= $donnees{$clef};
    print "$clef === $val
\n";
}
```