

## Unité d'enseignement RSX102 - Année 2025-26

### Technologies pour les applications en réseau : contribution au profil NetDevOps

Les cours et quelques TP sont assurés par François Lacomme (FL).

Les TP sont assurés en grande partie par Philippe Palau (PP).

## RSX 102 : Plan prévisionnel des cours et TP

---

### Préface :

1. Cours (FL) - Rappel : les couches supérieures du modèle OSI

### TP - (PP)

Installation d'une machine virtuelle Ubuntu (ou vérification du bon fonctionnement d'une MV Linux)

Configuration des clients Linux et Microsoft

## Chapitre 1 : Introduction

1. Définitions : Client-serveur ; P2P

## Chapitre 2 : Applications client-serveur dans Internet

*Hors programme. En annexe du cours :*

### 1. DNS - Domain Name System (En annexe)

- 1.1. Introduction
- 1.2. Service de noms
- 1.3. Noms de domaines
- 1.4. FQDN : Fully Qualified Domain Name
- 1.5. Enregistrer un nom de domaine
- 1.6. Enregistrement de ressources
- 1.7. Résolution de noms
- 1.8. Commandes et outils
- 1.9. DoH ; DNS over HTTPS

### TP - (PP) (à voir)

*/etc/hosts ; nslookup ; dig ; whois*

Installation et configuration de BIND sous Debian ; named ; dépannage et diagnostics DNS

## 2. Le web

- 2.1. Introduction ; les standards de base du web
- 2.2. URL ; URI
- 2.3. HTTP, HTTPS et HTTP/3
- 2.4. HTML et CSS
- 2.5. Javascript
- 2.6. Ajax
- 2.7. CGI ; Scripts coté serveur

### TP - (PP)

Installation et configuration du serveur HTTP Apache 2 sur Debian 12 ou 13 ;  
Configuration d'un site web ; Sites web virtuels  
Installation et configuration de Caddy.

### TP - (FL)

Utilisation de FTP et SSH.  
*Examen de requête et réponse HTTP ; scripts coté client et coté serveur*  
*Un minimum d'HTML5.*

## 3. XML ; Extensible Markup Language

- 3.1. Introduction
- 3.2. Utilisation de XML
- 3.3. Exploitation de document XML

## 4. JSON ; JavaScript Object Notation

### TP - (PP ou FL)

(PP ou FL) Exploitation et modification de documents XML et Json  
(FL) TP et démonstrations sur Ajax et JQuery

## 5. Services web

- 5.1. Introduction
- 5.2. SOAP (Simple Object Access Protocol)
- 5.3. REST (REpresentational State Transfer)

### TP - (FL)

Services web ; REST

## 6. SSL-TLS

- 6.1. SSL
- 6.2. TLS

*Hors programme. En annexe du cours :*

## 7. ~~Courrier électronique~~

- 7.1. ~~Définitions~~
- 7.2. ~~Composition d'un message (RFC 822)~~
- 7.3. ~~Transfert de messages~~
- 7.4. ~~Remise de messages~~
- 7.5. ~~Logiciels liés au courrier électronique~~

## 8. FTP ; File Transfer Protocol

## 9. NFS ; Network File System

- 9.1. Présentation
- 9.2. Autres systèmes de fichiers réseaux

## Chapitre 3 : Protocoles de transport

### 1. TCP

### 2. UDP

### 3. QUIC,

### 4. MPTCP

### 5. socket

- 5.1. Sockets
- 5.2. Web Sockets

### TP - (PP)

Programmation de sockets

### TP - (PP)

Protocole SNMP ; Analyse de trames (Wireshark)

## Chapitre 4 : Architectures client-serveur

### 1. Eléments d'architecture

### 2. Critères de comparaison

- 2.1. Données transportées

- 2.2. Types de dialogues
- 2.3. Structure et localisation des applicatifs
- 3. Type d'architecture (**à revoir**)
  - 3.1. Client-serveur à client passif
  - 3.2. Client-serveur de données
  - 3.3. Client-serveur distribué
  - 3.4. Client-serveur à objets distribués
  - 3.5. Client-serveur web
  - 3.6. Architectures à code mobile
  - 3.7. Architectures peer-to-peer
- 4. Registre distribué & blockchain
  - 4.1. Registre distribué
  - 4.2. blockchain
- 5. Rôle d'un serveur et serveur virtuel
  - 5.1. Services et fonctions
  - 5.2. VPS, Virtual Private Server
- 6. **Cloud computing : l'informatique dans les nuages**
  - 6.1. Introduction
  - 6.2. Virtualization + pay as you go + self service
  - 6.3. Les formes de cloud computing
  - 6.4. Les modèles de service. SaaS, PaaS, IaaS...
  - 6.5. Acteurs et applications
  - 6.6. Exemple du Cloud Microsoft Azure
- 7. **Introduction aux problématiques de partage de données et de tolérance aux fautes pour le Cloud :**
  - 7.1. *Exécutions en mode messages : Algorithmique distribuée, Ordre, Etat global, Coupes Cohérentes*
  - 7.2. *Partage de données répliquées et passage à l'échelle des architectures distribuées : cohérences uniformes, cohérences hybrides, cohérences faibles (Eventual Consistency, conjecture de Brewer et théorème CAP)*
  - 7.3. *Algorithmique distribuée en contexte tolérant aux fautes : types de pannes, consensus, auto-stabilisation*
- 8. **Introduction aux architectures d'applications sur le Cloud**
  - 8.1. *Exécution : Architectures orientées Micro-services*
  - 8.2. *Stockage : NFS, stockage large échelle (voir Ch.2 / 9 et ch. 6)*
  - 8.3. *Exemple du Cloud Microsoft Azure (le fournisseur de Cloud peut varier) (voir 6.6)*

## 9. Le middleware de base

- 9.1. Introduction
- 9.2. Service d'annuaire ; LDAP
- 9.3. Appel de procédure distante : RPC et gRPC
- 9.4. MOM
- 9.5. *Modèle Publish/Subscribe des Files de messages : MQTT, RabbitMQ, ZeroMQ, Kafka*

*Hors programme.*

## ~~Chapitre 5 : Internet des objets.~~

- 1. ~~Généralités~~
- 2. ~~Types de réseaux~~
- 3. ~~Définitions~~
- 4. ~~Protocoles de communication~~
  - 4.1. ~~CoAP~~
- 5. ~~Protocoles de routage~~
- 6. ~~Acteurs~~

## Chapitre 6 - Systèmes distribués

- 1. Introduction
- 2. Partage de données dans le cloud
  - Problématique ; Définitions ; Big data ; La cohérence de données (data consistency) ; Théorème CAP, alias théorème de Brewer
- 3. Hadoop

*Hors programme*

## ~~Chapitre 7 : Streaming et vidéos~~

- 1. ~~MPEG~~
- 2. ~~Streaming~~
- 3. ~~DASH~~
- 4. Prot. applicatif (automate ; en java)
- 5. web service REST
- 6.