

# Projets réseaux java

L3 informatique

2013/2014

## Remarques

Vous pouvez proposer vous-même un projet. Dans ce cas il faut m'envoyer sa description et s'il est intéressant vous pourrez le faire.

Le point de départ pour vos recherches doit être Wikipedia (<http://www.wikipedia.org>) et ensuite vous pouvez raffiner vos recherches en Google. Si vous avez de questions, n'hésitez pas de les poser par email ou personnellement.

Beaucoup de projets consistent dans l'implantation des protocoles Internet bien connus. Dans ce cas il faut impérativement consulter les RFC correspondants et implanter au moins une fonctionnalité minimale. Il est clair que vos applications doivent respecter les standards, le test s'effectuera donc avec d'autres applications implantant le protocole correspondant.

Pour la plupart des projet vous ne pouvez pas utiliser des bibliothèques existantes (ou du code existant) sur le net qui implémentent la totalité ou une partie des fonctionnalités demandés dans le projet. En cas de doute, n'hésitez pas à me contacter.

Je vous rappelle que vous devez me rendre le projet sous une forme d'un fichier .zip contenant un rapport (décrivant le projet, les membres, la documentation, les diagrammes UML et les détails d'implantation), les fichiers source et compilés. Vous pouvez utiliser n'importe quel outil de programmation, mais le code source doit pouvoir être compilé avec JDK en ligne de commande.

Les projets marqués par \* peuvent demander un travail de documentation ou de programmation important.

## Liste des projets

Projet 1 : Décryptage distribué des codes DES/RSA	3
Projet 2 : * Bittorrent	4
Projet 3 : DNS	5
Projet 4 : * Messagerie XMPP	6
Projet 5 : IRC	7
Projet 6 : Radio Internet (Shoutcast)	8
Projet 7 : * Client SNMP	9
Projet 8 : Backgammon	10
Projet 9 : * Simulateur des trains/métro	11
Projet 10 : * Serveur DHCP	12
Projet 11 : Bataille navale	13
Projet 12 : Snakes	14
Projet 13 : Tunnel HTTP	15
Projet 14 : * Protocol Buffers	16
Projet 15 : Scrabble	17
Projet 16 : POP3	18
Projet 17 : Client IMAP	19
Projet 18 : * Serveur FTP	20
Projet 19 : Mortal Kombat	21
Projet 20 : Partage de fichiers	22
Projet 21 : Loto	23
Projet 22 : Serveur HTTP+WebDAV	24
Projet 23 : * Client POP3 mobile	25
Projet 24 : Bataille navale mobile	26
Projet 25 : Dame de pique	27

## Projet 1: Décryptage distribué des codes DES/RSA

Les codes DES ou RSA demandent beaucoup de ressources pour leur décryptage. C'est pourquoi on propose d'écrire un système client/serveur qui permettra de décrypter une clé DES ou RSA en distribuant la tâche à plusieurs ordinateurs. Le serveur doit distribuer la tâche aux clients et interpréter les résultats.

### Détails

On suppose que vous êtes familier avec ces deux algorithmes. Sinon, vous pouvez regarder en premier temps wikipedia ([www.wikipedia.org](http://www.wikipedia.org)) et Google. Si vous en aviez besoin, on pourrait discuter le fonctionnement de ces algorithmes personnellement. Je vous rappelle qu'on utilise le DES standard 64bits (sans extensions ni 3DES) et le RSA sur 512/1024 bits.

Pour DES il faut décrypter le message texte, pour RSA il faut retrouver la clé privé à partir de la clé publique. Je vous suggère d'utiliser l'approche "brute force", car elle est la plus simple à programmer. Si vous avez le temps, vous pouvez implanter d'autres approches (cela sera apprécié pour la note).

La répartition des tâches doit tenir en compte le fait que les tâches doivent être faisables et ne pas prendre beaucoup de temps aux clients. Le principe est simple : si le client termine sa tâche et veut encore une, il la redemande. La quantité des clés à tester peut être déterminée expérimentalement. Il faut prévoir également le cas lorsque le client ne termine pas le calcul pour différentes raisons. Dans ce cas, il faut considérer la tâche comme non-faite. Vous pouvez vous inspirer de nombreux projets de décryptage de RSA existants.

## Projet 2: \* Bittorrent

Bittorrent est un protocole peer-to-peer d'échange des fichiers. On propose de réaliser un client et serveur Bittorrent ayant des fonctions basiques.

### Détails

Il faut réaliser en premier temps un client bittorrent basique qui doit permettre l'échange des fichiers et l'enregistrement sur le tracker. Ensuite, il faut réaliser un tracker simple. Vous pouvez vous inspirer des projets existants.

Liens :

<http://en.wikipedia.org/wiki/Bittorrent>

<http://www.bittorrent.com/documentation.html>

## Projet 3: DNS

DNS est un protocole de résolution des noms d'ordinateurs (en adresses IP) largement utilisé en Internet. Il est proposé de réaliser un serveur DNS restreint (A/PTR). À réaliser de préférence la version UDP.

### Détails

Vous trouverez beaucoup d'information concernant le fonctionnement du protocole DNS. Cependant n'oubliez pas que c'est un protocole binaire. N'hésitez pas à me contacter si vous seriez en difficulté. Il faut réaliser un serveur DNS, mais on se restreindra uniquement au serveur utilisant UDP (ou TCP) et qui pourra répondre uniquement aux requêtes A et PTR.

Liens :

<http://en.wikipedia.org/wiki/Dns>

## Projet 4: \* Messagerie XMPP

XMPP est un protocole utilisé pour créer des messageries instantanées. Créer une messagerie instantanée fondée sur XMPP. Tester avec les réseaux Jabber et GoogleTalk.

### Détails

Retrouvrez la spécification de XMPP. Il est probablement mieux de commencer la connaissance du protocole à l'aide d'un client existant (Jabber, GAIM). Vous devez faire la partie client du protocole et implanter uniquement le transfert des messages. Ensuite, si vous avez le temps, on vous propose de créer une petite extension et de l'implanter sous une forme de serveur ou gateway. Il est clair qu'il faut utiliser des outils standard pour le cryptage et le parsing du XML.

Liens :

<http://en.wikipedia.org/wiki/Jabber>

<http://en.wikipedia.org/wiki/XMPP>

[http://en.wikipedia.org/wiki/Google\\_Talk](http://en.wikipedia.org/wiki/Google_Talk)

## Projet 5: IRC

IRC est le plus ancien protocole pour des chats. Planter en Java un client et un serveur IRC.

### Détails

Le protocole est bien connu. Suivez bien le RFC. Pour le serveur faites une implantation basique : channels, modes principaux, dcc (mais pas de join et netsplit – donc un seul serveur).

Liens :

[http://en.wikipedia.org/wiki/Internet\\_Relay\\_Chat](http://en.wikipedia.org/wiki/Internet_Relay_Chat)

## Projet 6: Radio Internet (Shoutcast)

Planter un serveur qui permettra la diffusion de la musique par Internet (radio Internet) à l'aide du protocole SHOUTcast.

### Détails

Le serveur distribuera un flux (stream) de musique en format mp3 ou ogg. Vous devez utiliser le protocole SHOUTcast pour diffuser le flux. Il faut prévoir une interface graphique qui permettra de faire la programmation de la station (c.-à-d. ajouter/supprimer/déplacer les émissions). Le test se fera avec Winamp et WMP.

Liens :

[http://en.wikipedia.org/wiki/Internet\\_radio](http://en.wikipedia.org/wiki/Internet_radio)  
[http://en.wikipedia.org/wiki/Streaming\\_media](http://en.wikipedia.org/wiki/Streaming_media)  
<http://en.wikipedia.org/wiki/SHOUTcast>  
<http://en.wikipedia.org/wiki/Icecast>



## Projet 7: \* Client SNMP

SNMP est un protocole qui permet de demander des informations aux équipements connectés en réseau. Il faut réaliser un client SNMP.

### Détails

SNMP est un protocole binaire relativement simple. La difficulté du projet réside dans la lecture et la compréhension de la documentation. Il faut implanter la version 2 du protocole. Il serait envisageable d'intégrer la gestion des MIBs, plus particulièrement les MIBs de HP.

Liens :

[http://en.wikipedia.org/wiki/Simple\\_Network\\_Management\\_Protocol](http://en.wikipedia.org/wiki/Simple_Network_Management_Protocol)

<http://www.cuddletech.com/articles/snmp/>

<http://abcdrfc.free.fr/rfc-vf/rfc1157.html>

<http://www.snmplink.org/>

## Projet 8: Backgammon

Réaliser le jeu Backgammon en réseau ou on joue contre le temps.

### Détails

Ce jeu est très connu. Il faut réaliser la variante réseau et chronométrer le temps. Trouver un mécanisme permettant de gérer le temps correctement, même si les deux horloges des ordinateurs ne sont pas identiques.

Liens :

<http://en.wikipedia.org/wiki/Backgammon>

## Projet 9: \* Simulateur des trains/métro

Écrire un simulateur d'un réseau ferré (train/métro). Des compétences graphiques sont vivement conseillées.

### Détails

La structure de l'application est la suivante :

- Des ordinateurs contenant des morceaux du réseau. Les morceaux possèdent des points d'entrée/sortie. Chaque ordinateur est responsable de la simulation du passage des trains dans sa zone.
- Un serveur central qui gère l'interconnexion des zones ferrées.

Initialement le serveur central découvre les morceaux du réseau. Ensuite, on connecte certains points d'entrée/sortie des zones. Ensuite le serveur répartit les trains. Chaque zone gère individuellement ses trains (en les déplaçant d'une manière autonome d'un point à l'autre), mais le serveur doit pouvoir à tout moment obtenir l'image globale du réseau. Lorsqu'un train passe d'une zone à l'autre, le serveur central renvoie le train à l'ordinateur correspondant. Prévoir la possibilité de modifier les conditions du trafic en se connectant sur les ordinateurs de zone (pour modifier le réseau) ou sur le serveur central (pour modifier les paramètres de passage des trains, destinations etc).

## Projet 10: \* Serveur DHCP

DHCP est un protocole qui permet d'attribuer des adresses IP aux ordinateurs. Il faut réaliser un serveur DHCP.

### Détails

Vous pouvez écrire un client spécial pour faciliter les tests.

Liens :

[http://en.wikipedia.org/wiki/Dynamic\\_Host\\_Configuration\\_Protocol](http://en.wikipedia.org/wiki/Dynamic_Host_Configuration_Protocol)

## Projet 11: Bataille navale

Réaliser le jeu Bataille navale en version équipe ou il y a plusieurs équipes qui s'affrontent (au moins 3).

### Détails

Il s'agit du jeu classique, avec 2 adversaires. En version équipe, il y a  $n$  équipes. On peut voir les coups des autres équipes, et on peut jouer un coup contre une équipe quelconque. Le choix du coup se fait par consensus entre les membres de l'équipe, ou le cas échéant imposé par le capitaine.

## Projet 12: Snakes

Réaliser le jeu Snakes – le jeu classique avec le serpent ou plusieurs serpents appartenant aux joueurs différents évoluent sur le même plateau.

## Projet 13: Tunnel HTTP

Réaliser un programme qui permettra de retransmettre n'importe quelle connexion par le protocole HTTP (en cachant bien-sûr l'information). Tester sur l'exemple de SMTP (sur le serveur de l'université).

### Détails

En effet, le programme est composé de 2 parties :

1. Partie locale qui récupère une connexion (sur localhost) et la retransmet à la partie distante par HTTP.
2. Partie distante qui récupère l'information par HTTP de la partie locale et retransmet sur une connexion locale (sur localhost).

Bien sûr que les deux parties se trouvent sur des ordinateurs différents.

Regardez les liens et n'hésitez pas de me contacter.

Liens :

[http://en.wikipedia.org/wiki/Tunneling\\_protocol](http://en.wikipedia.org/wiki/Tunneling_protocol)

<http://www.nocrew.org/software/httpunnel.html>

## Projet 14: \* Protocol Buffers

Effectuer une implantation de Protocol Buffers de Google qui permettra de lire et de créer des messages (pour un nombre de types fini). Tester cette implantation pour commander à distance le lecteur de musique Clementine.

Liens :

<https://developers.google.com/protocol-buffers/docs/encoding>

<http://www.clementine-player.org/>



## Projet 15: Scrabble

Réaliser le jeu Scrabble. Implanter également la possibilité de voir comment jouent les autres sans participer au jeu.

### Détails

Ce jeu connu est présent parmi les jeux standard de Windows.

<http://fr.wikipedia.org/wiki/Scrabble>

## Projet 16: POP3

POP3 est un protocole de consultation des messages mail. Réaliser un serveur POP3 sécurisé (et un client qui va avec).

### Détails

Réaliser un serveur POP3 qui comprendra APOP et pourra également être utilisé en combinaison avec SSL/TLS.

Liens :

<http://en.wikipedia.org/wiki/Pop3>

[http://en.wikipedia.org/wiki/Secure\\_Sockets\\_Layer](http://en.wikipedia.org/wiki/Secure_Sockets_Layer)

## Projet 17: Client IMAP

IMAP est un protocole de consultation des messages mail. Réaliser un client IMAP.

### Détails

Liens :

<http://en.wikipedia.org/wiki/Imap>

## Projet 18: \* Serveur FTP

FTP est un protocole de transfert de fichiers. Réaliser un serveur FTP.

### Détails

Réaliser en premier temps un serveur utilisant le transfert passif, ensuite passer au transfert actif.

Liens :

<http://en.wikipedia.org/wiki/Ftp>

## Projet 19: Mortal Kombat

Il s'agit ici de réaliser un petit jeu de combat s'inspirant du jeu Mortal Kombat.

### Détails

En jeu, une fenêtre affiche les deux personnages, ainsi qu'un temps de round en haut de l'écran (se décrémentant toutes les secondes). De plus, les deux barres de vie des personnages sont représentées par des nombres initialisés à 100, visibles dans les deux coins supérieurs de la fenêtre de jeu. Chaque joueur contrôle son personnage, et peut le faire sauter (en diagonale ou verticalement), se déplacer à droite ou à gauche, s'accroupir, bloquer, et frapper. Deux coups sont disponibles : coup de pied, coup de poing. Chaque personnage dispose alors d'une palette de quatre coups (les coups diffèrent selon son état : accroupi ou debout). Lorsqu'une frappe est lancée, il faut, pour qu'elle touche : que le personnage ennemi soit à portée, qu'il ne bloque pas, et qu'il soit dans le bon état (un coup haut ne pourra pas toucher un personnage accroupi). Chaque attaque possède une portée et des dégâts définis.

Attention : Pour éviter la triche des joueurs c'est le serveur qui vérifiera tous les coups/déplacements.

[http://en.wikipedia.org/wiki/Mortal\\_Kombat](http://en.wikipedia.org/wiki/Mortal_Kombat)

## Projet 20: Partage de fichiers

Réaliser un serveur qui partagera certains fichiers et un client qui pourra accéder au partage (obtenir la liste de fichiers, transférer ou effacer des fichiers). Prévoir des interfaces graphiques, concevoir un protocole et gérer correctement plusieurs connexions.

### Détails

Vous pouvez utiliser n'importe quel protocole existant (excepté FTP) ou créer vous-même un protocole à votre guise. Le protocole SMB (Samba) est un bon exemple, mais vous n'êtes pas contraints à son implantation.

Liens :

[http://en.wikipedia.org/wiki/Server\\_Message\\_Block](http://en.wikipedia.org/wiki/Server_Message_Block)

## Projet 21: Loto

Réaliser le jeu du Loto. Prévoir plusieurs jeux en parallèle comportant plusieurs manches. Les cartons des joueurs doivent être valides dans tous les jeux.

Liens :

<http://fr.wikipedia.org/wiki/Loto>

## Projet 22: Serveur HTTP+WebDAV

Réaliser un serveur HTTP basique et le compléter avec le support du protocole WebDAV.

### Détails

Réaliser un serveur HTTP qui supporte tout l'ensemble des commandes HTTP et WebDAV. Par contre, le serveur ne doit pas forcément être performant ou optimisé.

Liens :

`http://en.wikipedia.org/wiki/Http`

`http://en.wikipedia.org/wiki/Webdav`



## Projet 23: \* Client POP3 mobile

Réaliser un client POP3 pour un téléphone portable. REMARQUE : vous devez posséder impérativement un téléphone portable compatible Android (me contacter et me fournir le modèle du téléphone avant de prendre le projet)!

### Détails

Pour la programmation vous pouvez utiliser NetBeans avec Mobility Pack, Eclipse avec le plugin mobile, ou directement le toolkit Android. Pour commencer le travail, lisez le tutoriel indiqué dans les liens ci-dessous. N'hésitez pas de me contacter.

Liens :

<http://developer.android.com/training/basics/firstapp/index.html>

<http://fr.openclassrooms.com/informatique/cours/creez-des-applications-pour-android>

<http://en.wikipedia.org/wiki/Pop3>

## Projet 24: Bataille navale mobile

Réaliser le jeu Bataille navale (version réseau) pour un téléphone portable. REMARQUE : vous devez posséder impérativement un téléphone portable compatible Android (me contacter et me fournir le modèle du téléphone avant de prendre le projet) !

### Détails

Pour la programmation vous pouvez utiliser NetBeans avec Mobility Pack, Eclipse avec le plugin mobile, ou directement le toolkit Android. Pour commencer le travail, lisez le tutoriel indiqué dans les liens ci-dessous. N'hésitez pas de me contacter.

Liens :

<http://developer.android.com/training/basics/firstapp/index.html>

<http://fr.openclassrooms.com/informatique/cours/creez-des-applications-pour-android>

## Projet 25: Dame de pique

Réaliser le jeu Dame de pique. Implanter également la possibilité de voir comment jouent les autres sans participer au jeu.

### Détails

Ce jeu connu est présent parmi les jeux standard de Windows.

[http://fr.wikipedia.org/wiki/Dame\\_de\\_Pique](http://fr.wikipedia.org/wiki/Dame_de_Pique)

<http://membres.lycos.fr/damedepique/>