



ESISAR

NE449 - Programmation répartie

TDM numéro 1

Table des matières

1 Objectifs du TDM numéro 1.....	1
2 Exercice 1 : Echange UDP avec l'outil netcat (avec une seule machine).....	1
3 Exercice 2 : Echange UDP avec l'outil netcat (avec deux machines).....	2
4 Exercice 3 : Envoi de datagramme UDP.....	2
5 Exercice 4 : Réception de datagramme UDP.....	2
6 Exercice 5 : Envoi / réception de datagramme UDP.....	2

1 Objectifs du TDM numéro 1

Le TDM1 va permettre la prise en main de l'API des sockets en JAVA, et réaliser des communications entre client et serveur avec UDP. Pour cela, vous devez utiliser Eclipse JAVA sous Linux.

2 Exercice 1 : Echange UDP avec l'outil netcat (avec une seule machine)

Lisez la documentation de l'outil netcat (dans un terminal, faire `man netcat`).

Echangez des messages par UDP depuis une console vers une autre console grâce à l'utilitaire netcat :

- sur une console, vous êtes en émission de datagrammes UDP vers un port particulier (par exemple : 2000)
- sur l'autre console, vous êtes en réception sur le port 2000

Observez avec wireshark les paquets IP échangés, et faites l'analogie avec ce que vous avez

vu en cours.

Question : l'ordre de lancement de netcat dans les deux consoles est-il important ? Pourquoi ?

3 Exercice 2 : Echange UDP avec l'outil netcat (avec deux machines)

Echangez maintenant des messages par UDP entre deux machines (faites alliance avec un binôme proche de vous!), toujours grâce à l'utilitaire netcat :

- sur une machine, vous êtes en émission de datagrammes UDP vers un port particulier (par exemple : 2000) et l'adresse IP de l'autre machine
- sur l'autre machine, vous êtes en réception sur le port 2000

Vous pouvez ainsi réaliser un système de « chat » entre binômes.

Observez avec Wireshark les paquets IP échangés.

4 Exercice 3 : Envoi de datagramme UDP

Réalisez un programme en Java qui envoie un datagramme UDP avec un message simple à l'intérieur (par exemple le mot « hello »), en vous basant sur les exemples fournis sur Chamilo, sur le port 2000 et à l'adresse 127.0.0.1.

Vérifiez le bon fonctionnement de votre programme en utilisant netcat pour réceptionner et afficher les datagrammes envoyés par votre programme.

5 Exercice 4 : Réception de datagramme UDP

Réalisez un programme en Java qui reçoit les datagrammes UDP sur le port 3000 et l'adresse 127.0.0.1, et qui les affiche à l'écran (en vous basant sur les exemples fournis en cours). Votre programme doit afficher le contenu du datagramme, et aussi l'adresse IP et le port de l'émetteur.

Vérifiez le bon fonctionnement en utilisant netcat pour envoyer des datagrammes vers votre programme, et vérifiez que l'affichage est bon.

6 Exercice 5 : Envoi / réception de datagramme UDP

Faites de nouveau alliance avec un binôme proche de vous, et faites communiquer votre programme de l'exercice 3 sur votre machine avec le programme de l'exercice 4 de l'autre binôme sur la machine de l'autre binôme.

Croisez ensuite en sens inverse.