

Group 17: Proposition de specification de protocole  
Application c2w

Abstract

Ce document presente la specification du protocole de communication utilise au sein de l'application c2w (Chat While Watching); Ce protocole permettra a tous les utilisateurs d'une meme salle (principal ou dee film) de s'echanger des messages.

Table of Contents

1. Introduction . . . . .	2
2. Types de paquet . . . . .	2
3. Format des Paquets . . . . .	3
3.1. Acquitement . . . . .	4
3.2. Erreur de sequence . . . . .	4
3.3. Demande de connexion . . . . .	5
3.4. Erreur de connexion . . . . .	6
3.5. Connexion Reussie . . . . .	6
3.6. Envoie liste des films . . . . .	7
3.7. Envoie Liste des users . . . . .	9
3.8. Avertissement Inactivite . . . . .	10
3.9. Avertissement SPAM . . . . .	10
3.10. Demande Acces film . . . . .	11
3.11. Envoie Message de Chat . . . . .	12
3.12. Quitter la MainRoom . . . . .	13
3.13. Quitter un film . . . . .	13
3.14. Message de deconnexion . . . . .	14
4. Fiabilite . . . . .	15
5. Exemple de scenario . . . . .	16
5.1. Scenario 1(sans erreur): Regarder un film . . . . .	16
5.2. Scenario 2(sans erreur): Chat dans une salle . . . . .	18
5.3. Scenario 3(sans erreur): Utilisateur inactif Dans la salle principale . . . . .	18
5.4. Scenario 4(sans erreur): Envoie de SPAM . . . . .	19
5.5. Scenario 5(avec erreur) : Quitter l'application . . . . .	22
6. Conclusion . . . . .	23
Authors' Addresses . . . . .	23

## 1. Introduction

Dans le cadre de l'application c2w : Chat While Watching, ce document decrit la specification du protocole de communication utilise pour la mise en place de ladite application. Cette application permet a un utilisateur, apres s'etre connecte, d'accéder a un certains nombres de videos, de pouvoir les regarder d'une part et, d'autre part, de pouvoir echanger des messages avec les utilisateurs se trouvant dans le meme salon.

Le serveur communiquera avec les utilisateurs aussi bien en utilisant le protocole de transport TCP que le protocole UDP. Les messages seront envoyes sous le format binaire.

Dans la suite du document nous specifions le fonctionnement du protocole, les types de messages envoyes, le format des differents messages et donnerons differents exemples de scenarii pouvant arriver dans l'execution du protocole.

## 2. Types de paquet

Ce tableau enumere l'ensemble des paquets qui seront utilises au sein de notre systeme. Leur description se fera dans la suite du document quand ils seront precises leurs formats de paquets.

code	TypeDePaquet	Sens de Transmission
0000	Acquittement	Client <-----> Serveur
0001	Demande de connexion	Client x-----> Serveur
0010	Erreur de connexion	Client <-----x Serveur
0011	Connexion Reussie	Client <-----x Serveur
0100	Envoi liste des films	Client <-----x Serveur
0101	Envoi liste de users	Client <-----x Serveur
0110	Avertissemnt Inactivite	Client <-----x Serveur
0111	Avertissemnt SPAM	Client <-----x Serveur
1000	Demande acces film	Client x-----> Serveur
1001	Erreur de sequence	Client <-----> Serveur
1010	Envoi message de Chat	Client <-----> Serveur
1011	Quitter la Main room	Client x-----> Serveur
1100	Quitter un salon de film	Client x-----> Serveur
1101	Message de deconnexion	Client <-----x Serveur

Figure 1

1. Client x-----> Server: Indique une communication du client vers le server
2. Client <-----x Server: Indique une communication du server vers le client
3. Client <-----> Server: Indique la possibilite d'avoir le cas 1. et 2.

										1										2										3									
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1								
+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-
	Code					Sequence											Taille																						
+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-	+	-

C'est la taille en bit de l'ensemble du message (c'est a dire de tous les champs du paquet reunis).

### 3.1. Acquitement

Ce paquet est transmis par le server ou le client pour informer le second acteur de transmission de la reception d'un paquet.

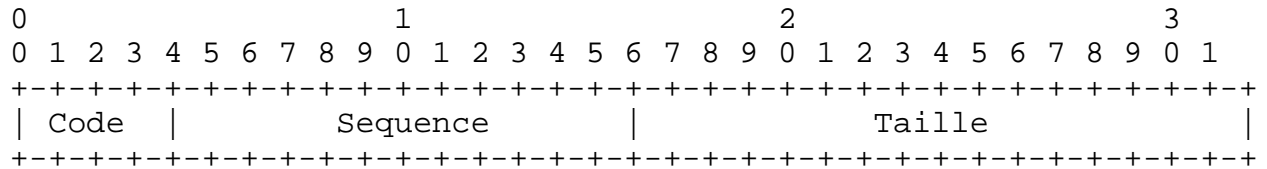


Figure 3

Code (4 bits): informant le client qu'il s'agit d'un acquitement.  
(Voir figure 1)

Sequence (12 bits): le numero de sequence du paquet.

Taille (16 bits): Taille totale du paquet.

### 3.2. Erreur de sequence

Ce message est envoye lorsqu'il y a contradiction entre la sequence attendue par le recepteur et celle recue. Dans notre systeme, chaque paquet est identifie de maniere unique par son numero de sequence. Lorsqu'un recepteur recoit plusieurs fois le meme numero de sequence, il detecte la redondance de message et soumet a l'emetteur cette erreur.

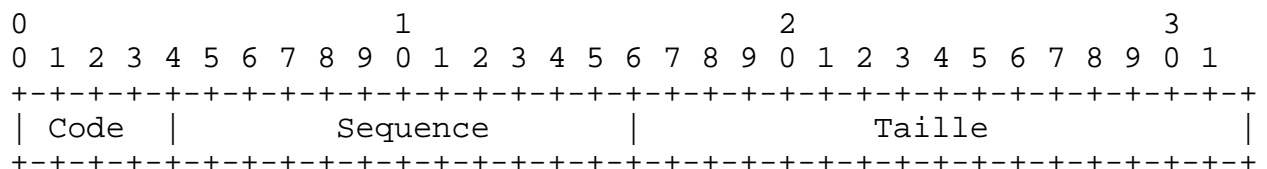


Figure 4

Code (4 bits): informant le client qu'il s'agit d'une notification d'erreur de sequence. (Voir figure 1)

Sequence (12 bits): le numero de sequence du paquet.

Taille (16 bits): Taille totale du paquet.

### 3.3. Demande de connexion

Ce paquet est le premier envoye par le client au server pour la demande de connexion.

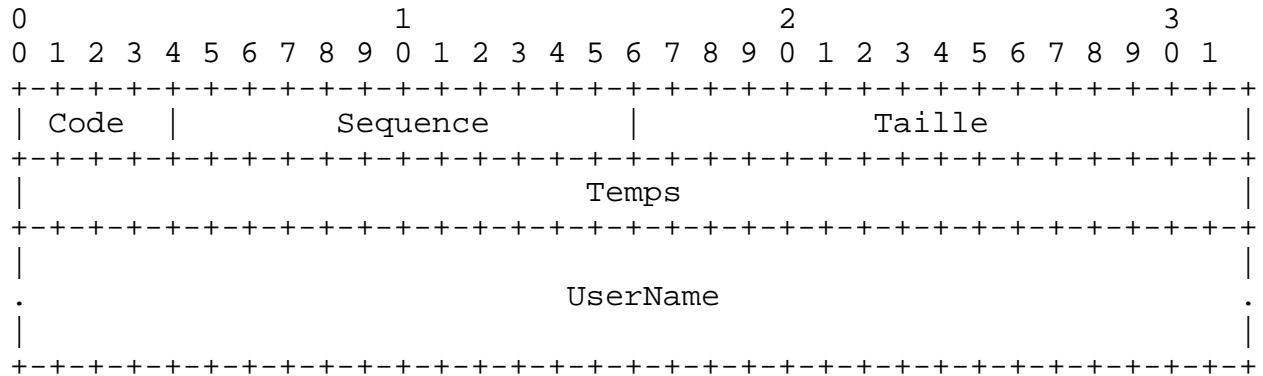


Figure 5

Code (4 bits): informant le server qu'il s'agit d'une demande de connexion a l'application. (Voir figure 1)

Sequence (12 bits): le numero de sequence du paquet.

Taille (16 bits): Taille totale du paquet.

Temps (32 bits)

Temps envoye par le client a chaque interaction volontaire avec le server: Demande de connection, envoie de message Chat, Quitter un film, Demande Acces film. Ce temps nous permet de mesurer le temps d'inactivite d'un client. Comme presnter dans le scenarion 3 (Voir 5.3) un client inactif pendant 15 min se fera systematique deconnecter lorsqu'il est dans la Main Room, les clients dans les salles de film n'etant pas concerner. Cela devrait permettre l'accès au service a d'autre client.

Username (256 bits)

Ce champs contient le Username du demandeur de connexion. Tout les caracteres UTF-8 sont acceptes. Le Username du client est code sur 256 bits et doit respecter certaines contraintes lors de la connexion d'un client.

1. Le username doit pouvoir etre code sur 256 bits
2. Le Username choisit ne doit pas etre deja utilise

3. Le username ne doit pas figure parmi une liste de username juges offensant Exemple : Hitler, Merde,...

#### 3.4. Erreur de connexion

Paquet envoye au demandeur de connexion lorsque son username est invalide.

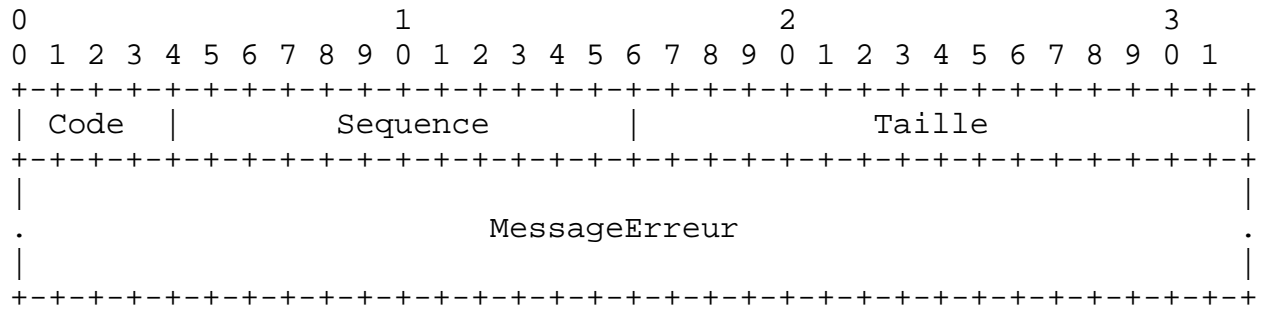


Figure 6

Code (4 bits): informant le client qu'il s'agit d'une notification d'erreur de connexion. (Voir figure 1)

Sequence (12 bits): le numero de sequence du paquet.

Taille (16 bits): Taille totale du paquet.

MessageErreur (1024 bits)

Ce champ contient un message a afficher lors d'une erreur de connexion. Le contenu du message permettra a l'utilisateur de savoir avec exactitude La cause l'erreur. Le message est encode en UTF-8. Suivant l'erreur sur le username le client saura que sont username est deja utilise , trop long ou n'est pas accepte.

#### 3.5. Connexion Reussie

Ce paquet est transmis par le server a un client lorsque la connexion a l'application est acceptee .

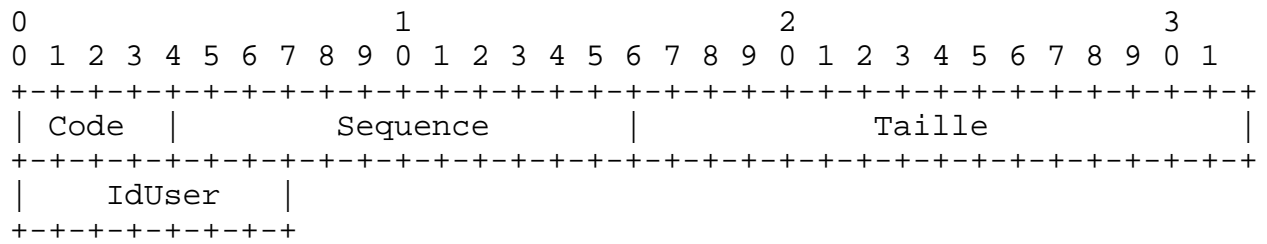


Figure 7

Code (4 bits): informant le client qu'il s'agit d'une notification de connexion reussie. (Voir figure 1)

Sequence (12 bits): le numero de sequence du paquet.

Taille (16 bits): Taille totale du paquet.

IdUser (7 bits)

Ce champ contient l'identifiant d'un user connecte. Code sur 7 bits un IdUser est attribue a chaque client lors de sa connexion et doit permettre son identification sans l'utilisation de son username afin de reduire la taille des paquets de communication. Le nombre maximal de User connecte est 128

### 3.6. Envoie liste des films

Ce paquet est transmis au client lors de sa premiere connexion (sauf ajout de nouveau film) pour ses futures selection de film a regarder.

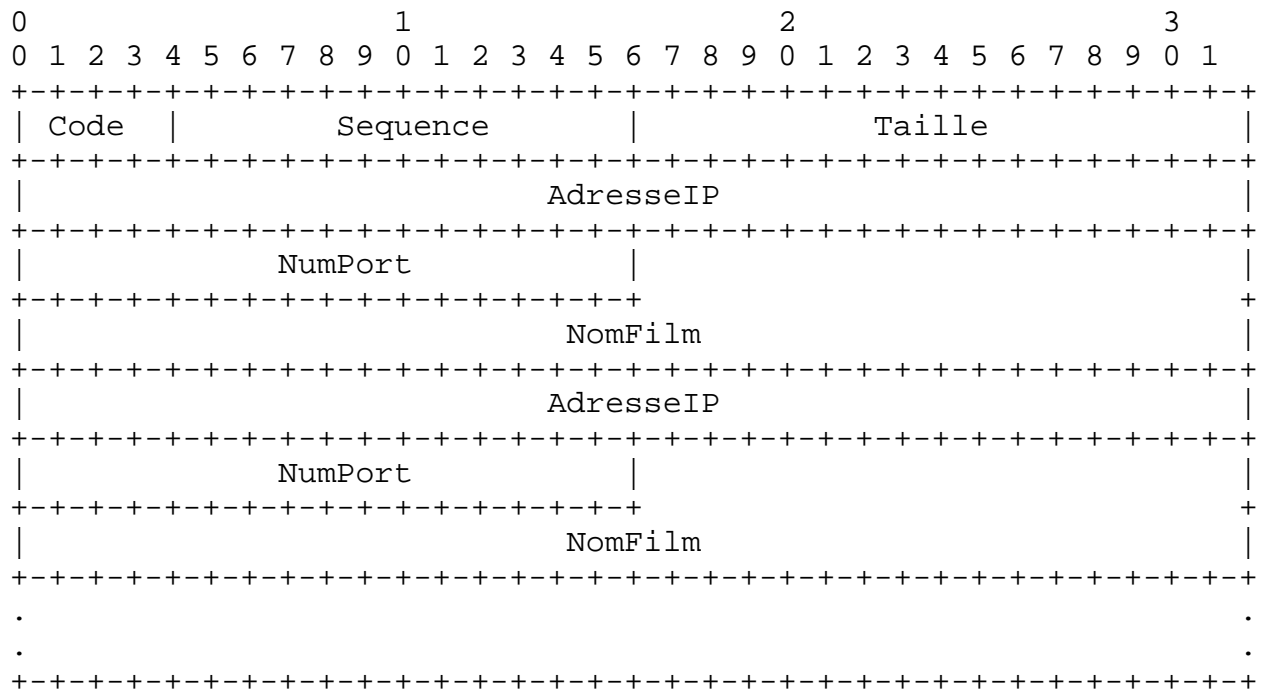


Figure 8

Code (4 bits): informant le client qu'il s'agit de la liste des films disponibles. (Voir figure 1)

Sequence (12 bits): le numero de sequence du paquet.

Taille (16 bits): Taille totale du paquet.

Adresse IP (32 bits)

Ce champ correspond a l'adresse Ip du films situe dans le champ Nomfilm qui suit.

NumPort (16 bits)

Ce champ correspond au Port utilise par le film situe dans le champ Nomfilm qui suit.

NomFilm (256 bits)

Ce champ contient Le nom d'un film. Tous les carateres UTF-8 doivent etre accepter et le message codes sur 256 bits. Le champ Salle (voir 3.10) etant code sur 4 bits l'application doit dispose d'au maximun 15 films, la 16e salle etant la MainRoom (Salle Principale)



### 3.7. Envoie Liste des users

Ce paquet est transmis par le server aux differents clients lorsqu'une personne rejoint ou quitte une salle (MainRoom ou Salle de film). Effectuant ainsi la mise a jour des clients presents dans chaque salle

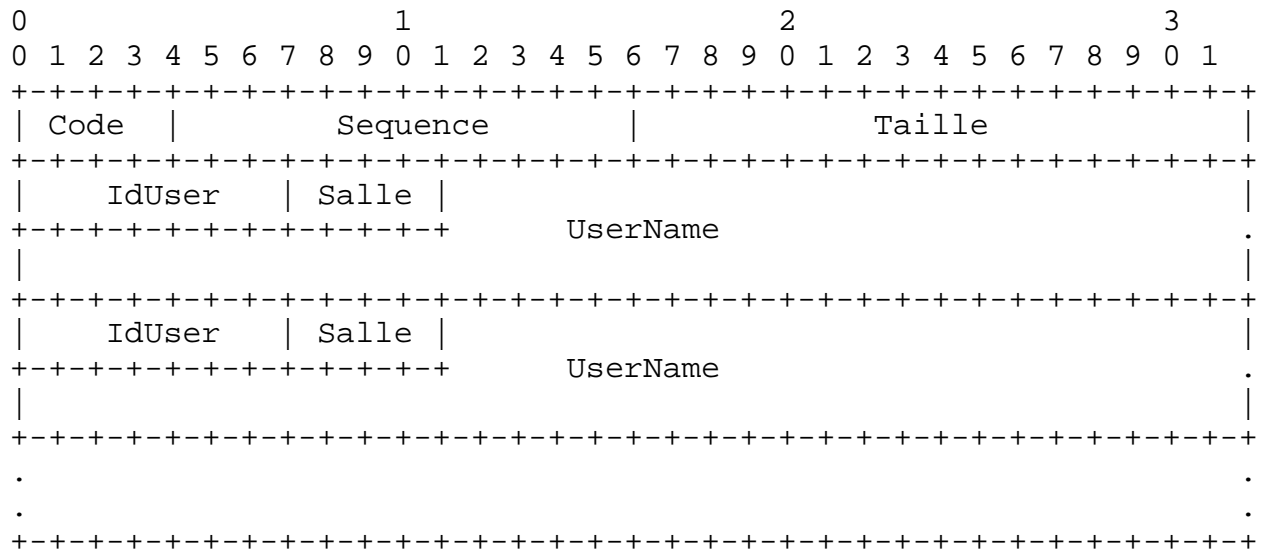


Figure 9

Code (4 bits): informant le client qu'il s'agit d'une liste de users.  
(Voir figure 1)

Sequence (12 bits): le numero de sequence du paquet.

Taille (16 bits): Taille totale du paquet.

IdUser (7 bits): l'identifiant du user dont le u sername se trouve dans le champ qui suit.

Salle (4 bits)

Ce champ code sur 4 bits contient le numero d'une Salle.  
0000 peut etre le numero pour designer la Main Room. Chaque Film est aussi represente par sa salle (numero de salle :xxxx !=0000) Le champ salle doit permettre de localiser un client a tout moment. En d'autre terme chaque un client est lier a une salle qui permet au server de connaitre les destinataires des messages qu'il lui envoie.

Username (256 bits): Username d'un client se trouvant dans une salle (MainRoom ,Salle de film). (Voir 3.3)

### 3.8. Avertissement Inactivite

Ce paquet est transmis a un client se trouvant dans la MainRoom lorsque aucune activite de sa part n'est detectees pendant un duree de 14 min.

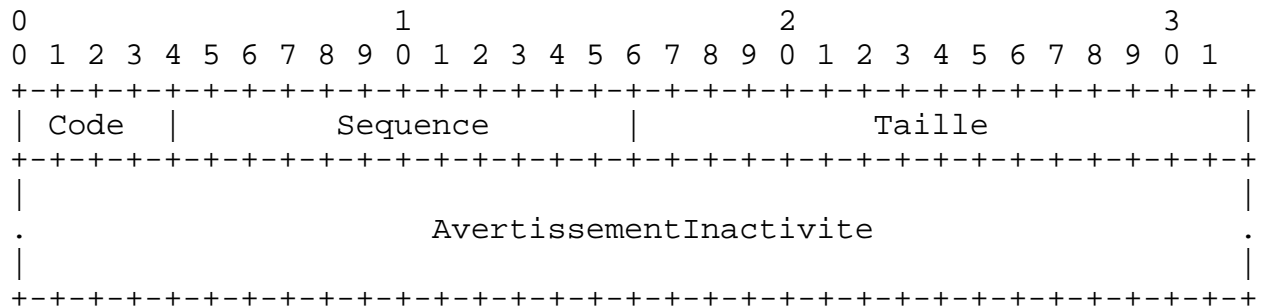


Figure 10

Code (4 bits): informant le client qu'il s'agit d'un message d'avertissemnt a afficher. (Voir figure 1)

Sequence (12 bits): le numero de sequence du paquet.

Taille (16 bits): Taille totale du paquet.

AvertissementInactivite (1024 bits)

Ce champ contient un message avertissant le client qu'il sera deconnecte dans 1 minute parce qu'il est inactif dans la MainRoom. Tous les carateres UTF-8 doivent etre accepter et le message codes sur 1024 bits Le message doit s'afficher parmi les messages de chat mais etre vu uniquement par le client concerne.

### 3.9. Avertissement SPAM

Ce paquet est transmis a un client se trouvant dans une salle quelconque lorsqu'il envoi plusieurs messages sur un delai tres court (2 messages en 1s). Le server identifie ces messages comme des SPAM.

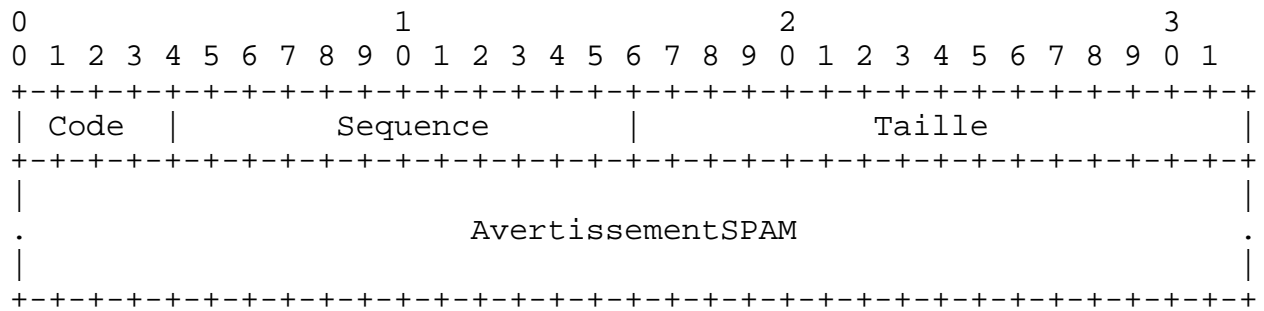


Figure 11

Code (4 bits): informant le client qu'il s'agit d'un message d'avertissement à afficher. (Voir figure 1)

Sequence (12 bits): le numero de sequence du paquet.

Taille (16 bits): Taille totale du paquet.

AvertissementSPAM (1024 bits)

Ce champ contient un message d'avertissement le client qu'il sera deconnecte dans au prochain avertissement pour envoi repete de message (2 messages par seconde). Tous les caracteres UTF-8 doivent etre acceptes et le message codes sur 1024 bits Le message doit s'afficher parmi les messages de chat mais etre vu uniquement par le client concerne.

### 3.10. Demande Acces film

Ce paquet est envoye par un client au server lorsqu'il veut joindre une salle de film.

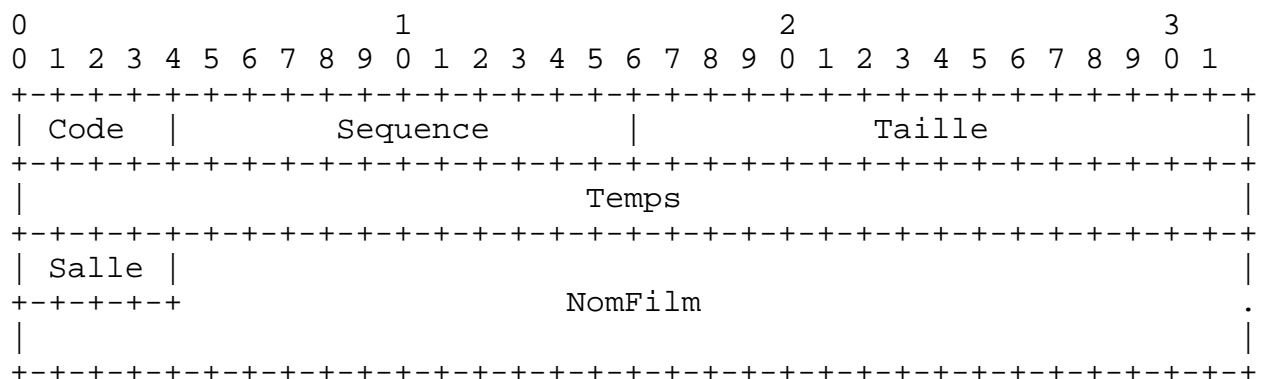


Figure 12

Code (4 bits): informant le server qu'il s'agit d'une demande d'accès a une salle de film. (Voir figure 1)

Sequence (12 bits): le numero de sequence du paquet.

Taille (16 bits): Taille totale du paquet.

Temps (32 bits): indiquant au server le temps d'envoi du paquet utilise pour le calcul du temps d'inactivite. Celle nouvelle valeur devient la valeur de reference. (Voir 3.3)

Salle (4 bits): contient la salle actuel du client (MainRoom salle=0000) pour l'avertissement des users de la presente salle.

NomFilm (256 bits): contient le nom du film souhaite par le client. (voir 3.6)

### 3.11. Envoie Message de Chat

Ce paquet contient les messages courants de discussion dans les salles. Il est transmis par le client au server et retransmis par le server a l'ensemble des clients concernes

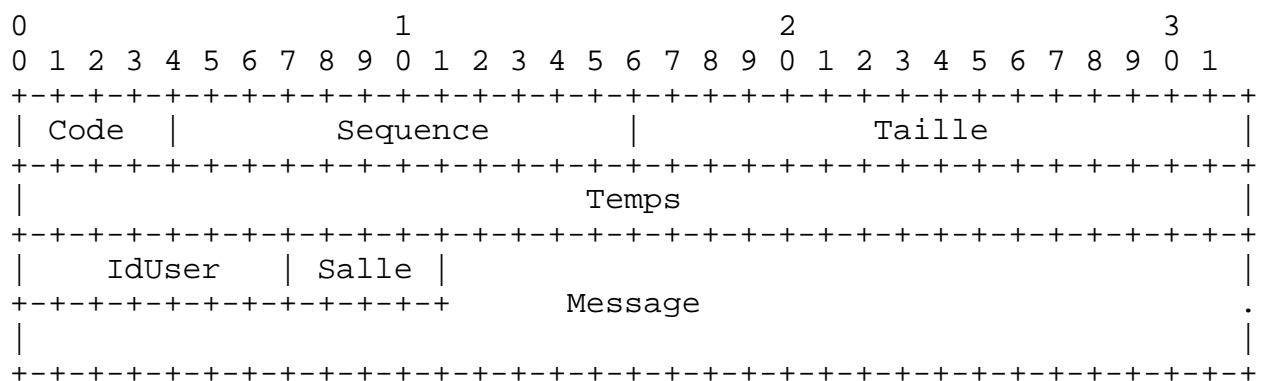


Figure 13

Code (4 bits): Informant le server qu'il s'agit d'un message chat a retransmettre. Informant le client que c'est un message a afficher (Voir figure 1)

Sequence (12 bits): le numero de sequence du paquet.

Taille (16 bits): Taille totale du paquet.

Temps (32 bits): indiquant au server le temps d'envoi du paquet utilise pour le calcul du temps d'inactivite. Celle nouvelle valeur devient la valeur de reference. (Voir 3.3)

IdUser (7 bits): l'identifiant du user a l'origine du message.

Salle (4 bits): la salle du user a l'origine du message.

Message (Taille variable)

Ce champ contient un message de chat envoye par un client en destination des clients presents dans la meme salle que lui. Tous les caracteres UTF-8 doivent etre accepter et la taille du message doit etre variable. Le client doit envoyer au plus un message par seconde et message s'afficher uniquement chez les clients de la salle de l'emetteur.

### 3.12. Quitter la MainRoom

Ce paquet est transmis par un client au server lorsqu'il souhaite quitter l'application.

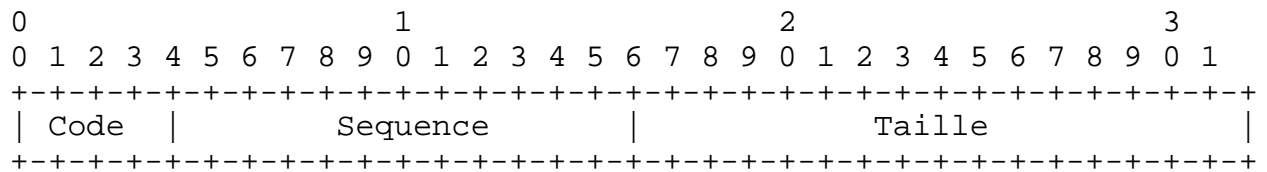


Figure 14

Code (4bits): informant le server que le client souhaite quitter l'application. (Voir figure 1)

Sequence (12bits): le numero de sequence du paquet.

Taille (16bits): Taille totale du paquet.

### 3.13. Quitter un film

Ce paquet transmis par le client au server lui signifaint qu'il souhaite quitter la salle dans laquelle il se trouve et rejoindre la MainRoom

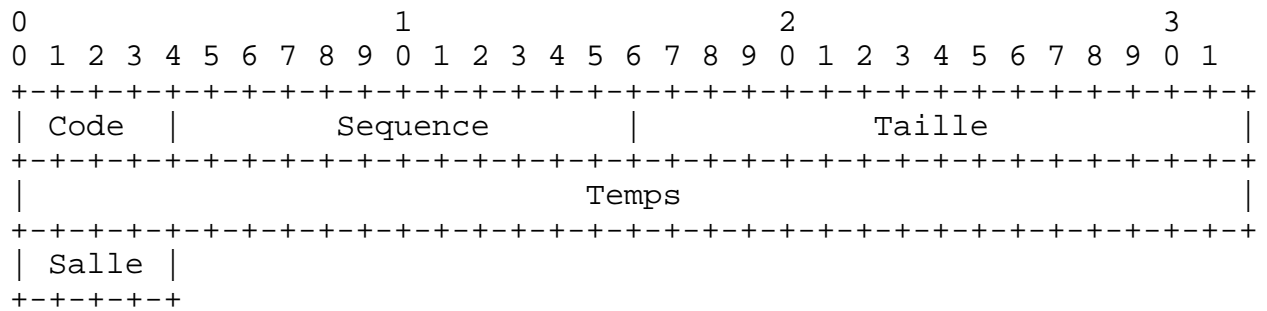


Figure 15

Code (4bits): informant le server qu'il souhaite rejoindre la Main Room. (Voir figure 1)

Sequence (12bits): le numero de sequence du paquet.

Taille (16bits): Taille totale du paquet.

Temps (32 bits): indiquant au server le temps d'envoi du paquet utilise pour le calcul du temps d'inactivite. Celle nouvelle valeur devient la valeur de reference. (Voir 3.3)

Salle (4 bits): Le client joint sa localisation au message pour que les clients de cette salle soient avertis de son depart

### 3.14. Message de deconnexion

Ce message est transmis au client par le server lorsque ce dernier deconnecte volontairement le client de l'application Soit pour inactivite ou pour un detection de SPAM.

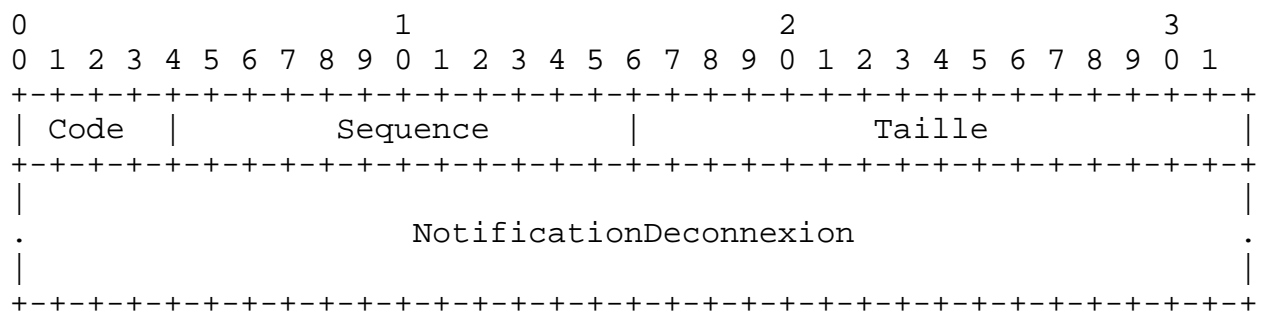


Figure 16

Code (4bits): informant le client qu'il s'agit d'un message a afficher. (Voir figure 1)

Sequence (12bits): le numero de sequence du paquet.

Taille (16bits): Taille totale du paquet.

NotificationDeconnexion (1024 bits)

Ce champ contient un message avertissant le client qu'il a ete deconnecte. Pour ce message, tous les caracteres UTF-8 doivent etre acceptes et le message codes sur 1024 bits. Le message doit s'afficher parmi les messages de chat mais etre vu uniquement par le client concerne.

#### 4. Fiabilite

La fiabilite du protocole est assuree par trois principaux elements : Le numero de sequence, l'acquittement et les messages d'erreur.

Le type de message acquittement: Ce type de message permet de s'assurer que les utilisateurs ou le serveur ait bien recu le message. Les deux entites attendent de recevoir ce message avant l'emission du prochain message meme message. En effet, en cas de non reception du message d'acquittement, l'emetteur considere que le recepteur n'a pas recu le message et le renvoie. De cette maniere, nous limitons le nombre de renvoi de message dans le systeme. Il est a noter qu'en outre, pour ne pas surcharger le serveur, les clients attendront toutes les secondes pour envoyer un message au serveur.

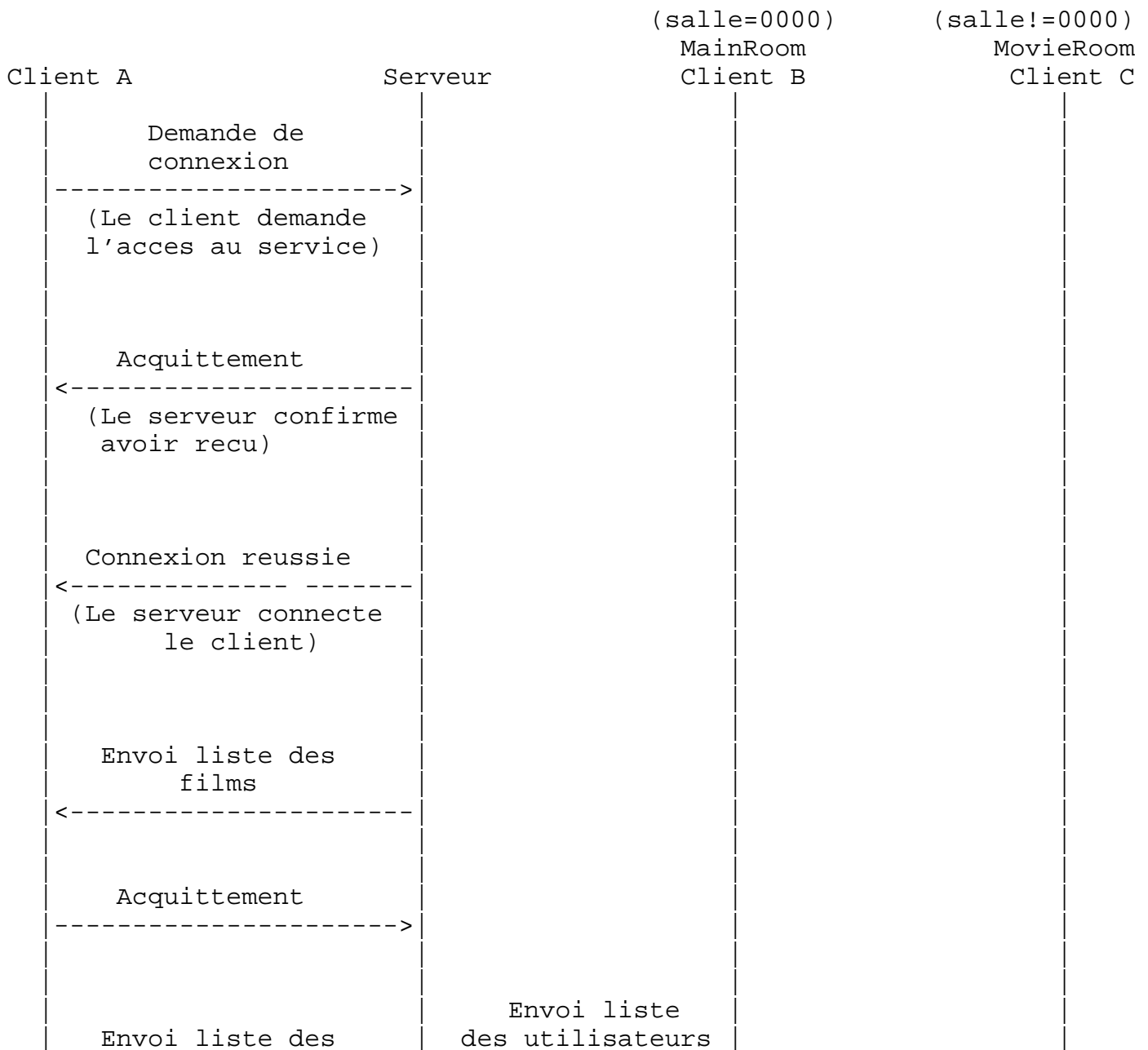
Le numero de sequence : Chaque message de notre service est associe a un numero de sequence, qui permet de l'identifier de maniere unique. De cette maniere, nous pouvons determiner si paquet recu est le bon. Prenons un exemple. Supposons que le client A veut communiquer avec le serveur. Il lui envoie le paquet de sequence 5, le serveur le recoit et lui envoie un acquittement. Malheureusement, le client ne le recoit pas. Le client renvoie donc le paquet de sequence 5 au serveur. Le serveur ayant deja recu ce paquet, signale une erreur au client en lui envoyant un paquet "Erreur de sequence". Le client A sait donc que le paquet de l'acquittement a ete egare, incremente son compteur de sequence et cesse les reemissions Cela s'averera tres utile dans le cas de la transmission sous le protocole UDP

Messages d'erreur : Les messages d'erreur envoyes lors de l'execution des taches du serveur permettront de reconnaitre les erreurs produites pas les clients afin de les corriger, sans toutefois avoir a se demander ce qui se passe.

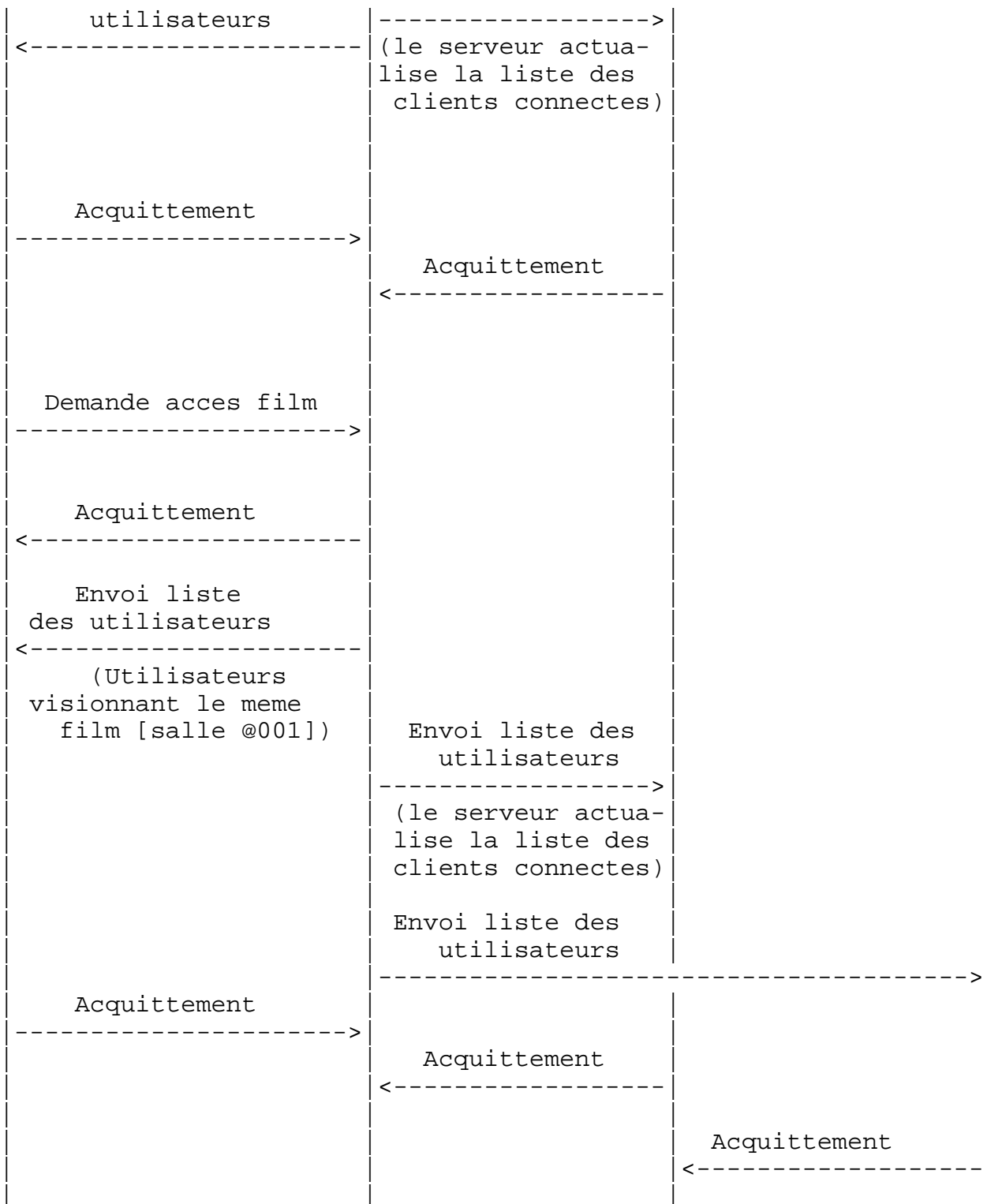
## 5. Exemple de scenario

## 5.1. Scenario 1(sans erreur): Regarder un film

Dans ce scenario, nous allons expliciter la suite de messages envoyee entre un client et le serveur lorsque l'utilisateur veut regarder un film sur la plateforme. Nous irons de la connexion au service jusqu'a l'accès au film.

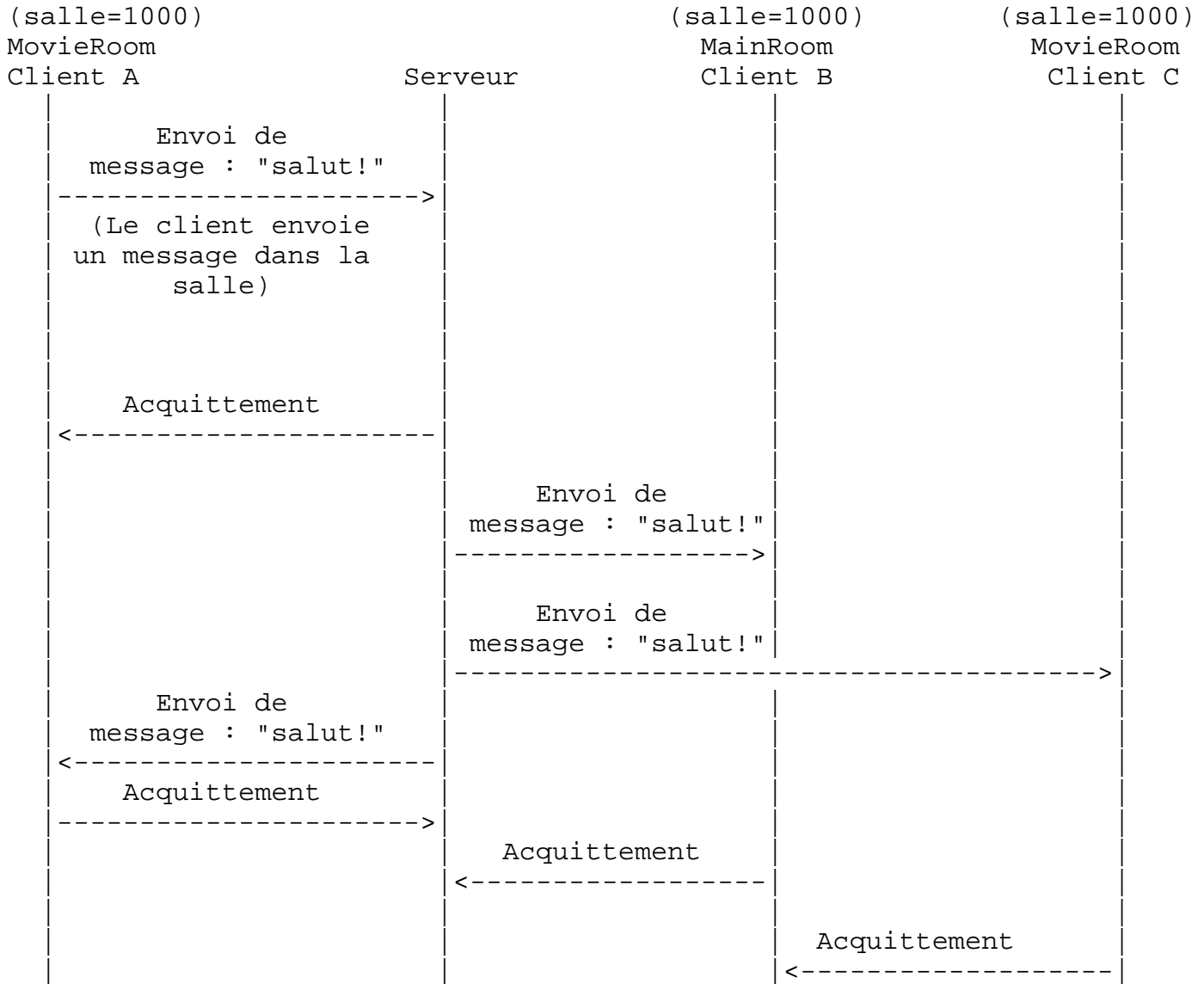






## 5.2. Scenario 2(sans erreur): Chat dans une salle

Ce scenario decrit la suite de message envoyes dans le cas d'un envoi de messages dans une meme salle.



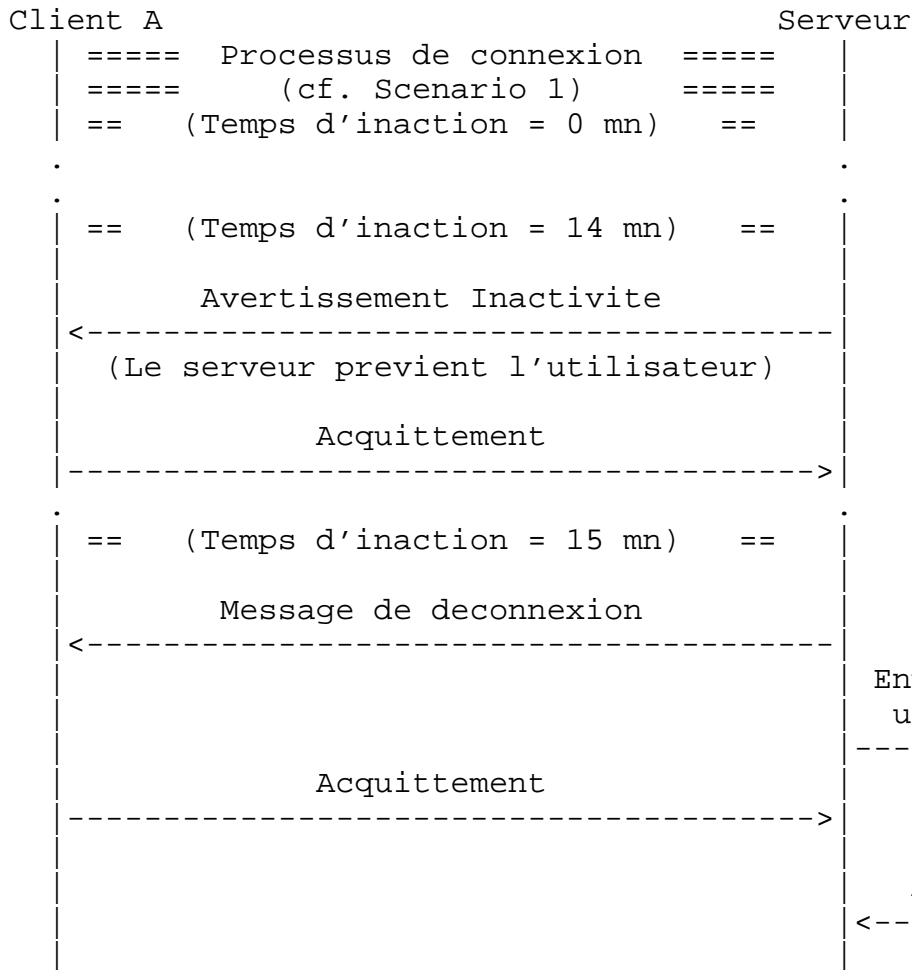
## 5.3. Scenario 3(sans erreur): Utilisateur inactif Dans la salle principale

Ce scenario decrit la suite de messages envoyes dans le cas ou un utilisateur est inactif dans la salle principale.

(salle=0000)

MainRoom

Client B



## 5.4. Scenario 4(sans erreur): Envoie de SPAM

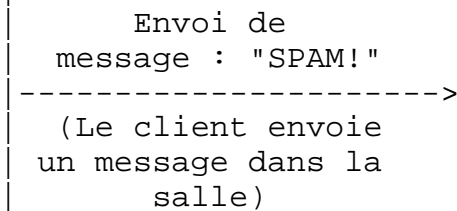
Ce scenario decrit l'ensemble des messages echanges lorsque l'utilisateur envoie des messages a repetition dans le chat dans le but de nuire (SPAM). Le serveur va donc le deconnecter.

(salle=1000)

MovieRoom

Client A

Serveur



(salle=1000)

MainRoom

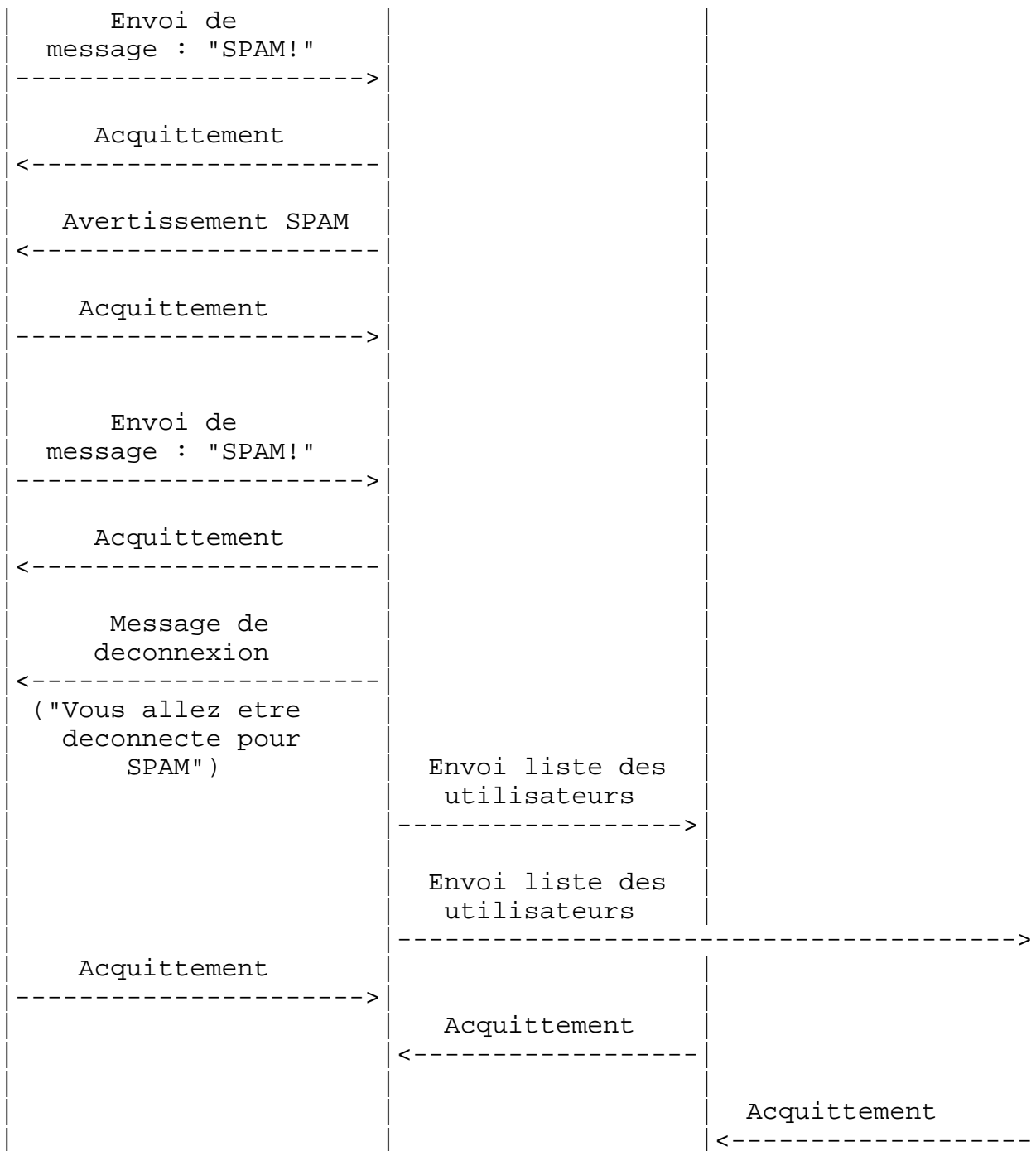
Client B

(salle=1000)

MovieRoom

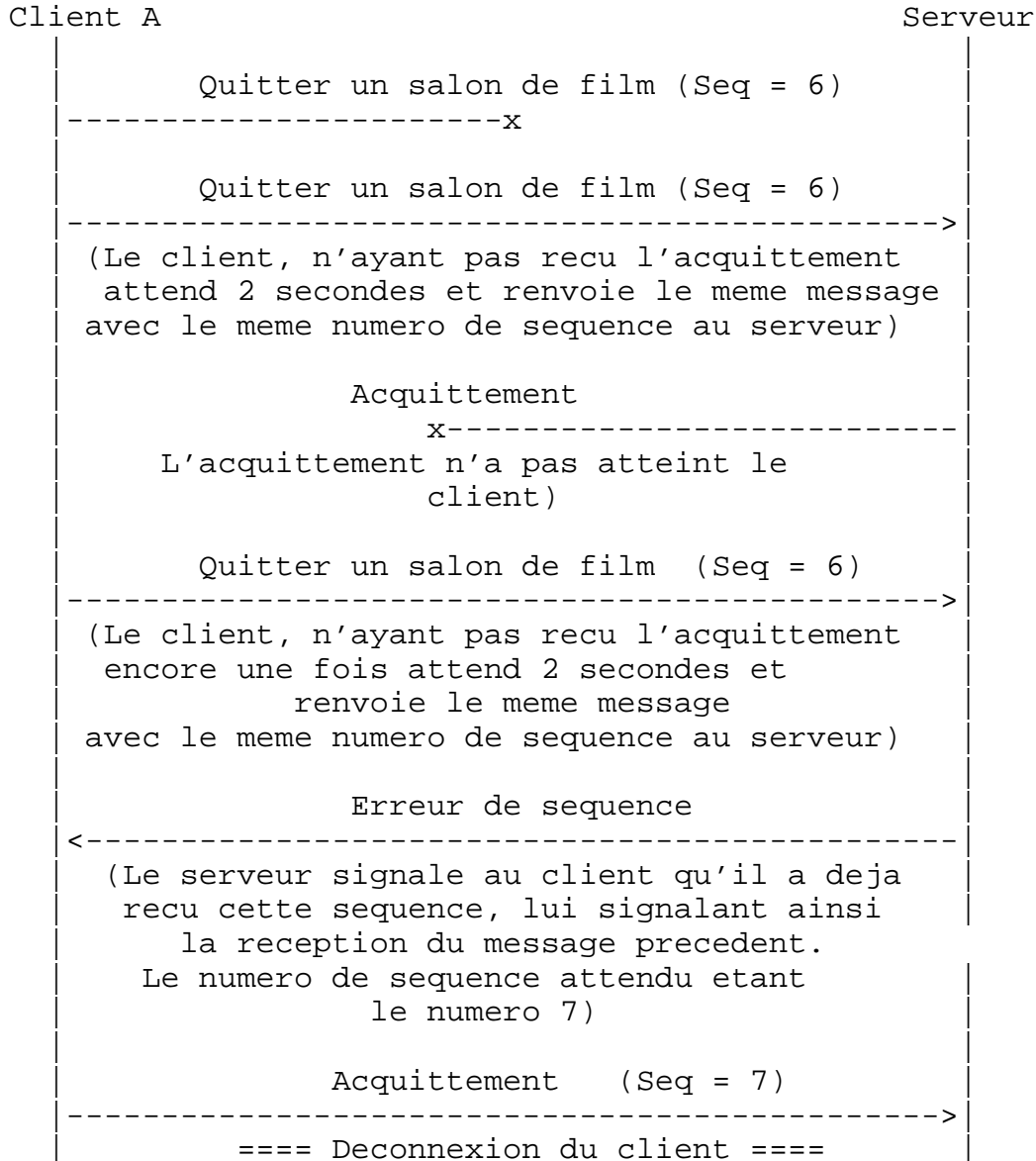
Client C





## 5.5. Scenario 5(avec erreur) : Quitter l'application

Dans ce scenario, nous allons simuler la suite de messages envoyes lorsque l'utilisateur veut quitter l'application. Cependant, nous ajouterons quelques erreurs qui peuvent arriver dans notre systeme et nous montrerons comment nous comptons les gerer. Sur la figure Seq designera le numero de sequence.



## 6. Conclusion

Chat While Watching est une application offrant aux utilisateurs un service de chat lorsque ceux-ci selectionnent le film qu'ils veulent regarder ou encore pendant le visionnage. En suivant le protocole enonce dans ce document, nous pensons que l'application peut garantir ce service de chat tout en gardant un certain confort et une bonne fiabilite.

### Authors' Addresses

Claude KADJO MAMI  
Telecom Bretagne  
655, Avenue du Technopole  
Brest, Bretagne 29280  
France

Email: [claude-yrvine-axel.kadjo-mami@imt-atlantique.net](mailto:claude-yrvine-axel.kadjo-mami@imt-atlantique.net)

Eric Xavier N'GUESSAN  
Telecom Bretagne  
655, Avenue du Technopole  
Brest, Bretagne 29280  
France

Email: [eric-xavier.nguessan@imt-atlantique.net](mailto:eric-xavier.nguessan@imt-atlantique.net)