

```

@startuml
title Envoi de messages entre un server et n clients
== Initialisation ==
actor Client
entity Server
control VerifSalon
actor AutresClients

activate Server
activate Client
Server -> Server : Initialisation du serveur, en attente de clients
Client -> Client : choix du pseudo
Client -> Server : connexion
deactivate Client

== Choix salon (thread_chan) ==

Server -> Client : envoi de la liste des salons
activate Client
Client -> Server : envoi du nom de salon à rejoindre
Server -> VerifSalon : vérification du salon (existence + nb max clients non atteint)
activate VerifSalon
VerifSalon -> Server : reponse VerifSalon
deactivate VerifSalon

loop "rep == salon choisi n'existe pas"
  Server -> Client : envoie message erreur + demande nouveau salon
  Server -> VerifSalon : vérification du salon (existence + nb max clients non atteint)
  activate VerifSalon
  VerifSalon -> Server : reponse VerifSalon
  deactivate VerifSalon
end

Server -> Client : client connecté au salon

== Conversation (thread_func) ==
loop infinie
  Client -> Server : envoi d'un message
  activate Client

  alt message != "fin" && message != "/file"
    Server -> AutresClients : transfert du message aux autres clients connectés au même
    salon que Client
    activate AutresClients
  else

```

```

alt message == "/file"
    note over Server : Des messages peuvent encore être envoyés par les autres clients
    durant la phase d'envoi de fichier
        Server --> Client : en attente du choix du fichier
        alt abandonne l'envoi de fichier
            Client -> Server : envoie "/abort" pour abandonner
        else continue l'envoi de fichier
            Client -> Client : Choix du fichier à envoyer
            Client -> Server : envoi du fichier
            Server -> AutresClients : transfert du fichier aux autres clients connectés au même
salon que Client
            deactivate AutresClients
        end
    else message == "fin"
        Server -> Client : Déconnexion du client
    end
end
end
@enduml

```