Etherclan - Rede distribuída descentralizada para localização de outros usuários

Henrique Gemignani Passos Lima <henrique@gemignani.org>, Orientador: Prof. Dr. Daniel Macêdo Batista
Instituto de Matemática e Estatística, Universidade de São Paulo - Trabalho de Conclusão de Curso

Introdução

Desenvolvedores de jogos eletrônicos livres que desejam implementar alguma funcionalidade multijogador sofrem com um problema: manter servidores, ou pelo menos uma lista desses. Jogos comerciais mais antigos para PC adotavam a estratégia de permitir que o jogador se conectasse a um endereço IP qualquer. Também era popular o jogo ter um lista de servidores oficial, que podia ser acessada de dentro do jogo. Já jogos comerciais recentes sempre possuem um servidor oficial principal do qual todas funcionalidades multijogador dependem.

No caso de um jogo livre, o desenvolvedor tem duas opções: manter ele mesmo um servidor principal ou deixar esse trabalho para a comunidade. O primeiro implica num custo de manutenção para o desenvolvedor, enquanto o segundo aumenta a barreira de entrada para novos jogadores, dado que o jogador precisa ativamente procurar tal lista.

Neste trabalho, propomos uma rede distribuída descentralizada na qual todos os jogadores e servidores participam como uma solução alternativa.

Objetivos

- Desenvolver uma rede peer-to-peer com o propósito de permitir que usuários do jogo consigam encontrar outros jogadores, sem a necessidade de um servidor central.
- Implementar pelo menos 1 prova de conceito.

Rede peer-to-peer

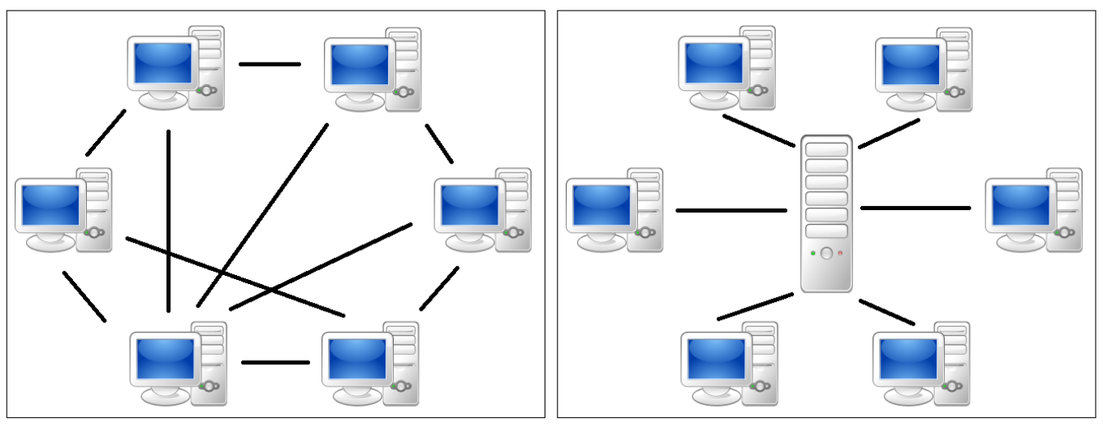


Figura: Comparação de rede Peer-to-peer (Esquerda) com Cliente-Servidor (Direita). Fonte: Baseado em figura de Wikimedia Commons

Protocolo de Rede

Quando mensagens são trocadas através de uma rede de computadores seguindo um sistema de regras digitais, tal sistema é chamado de um protocolo de rede. Sistemas de comunicação usam formatos bem-definidos para trocar mensagens.

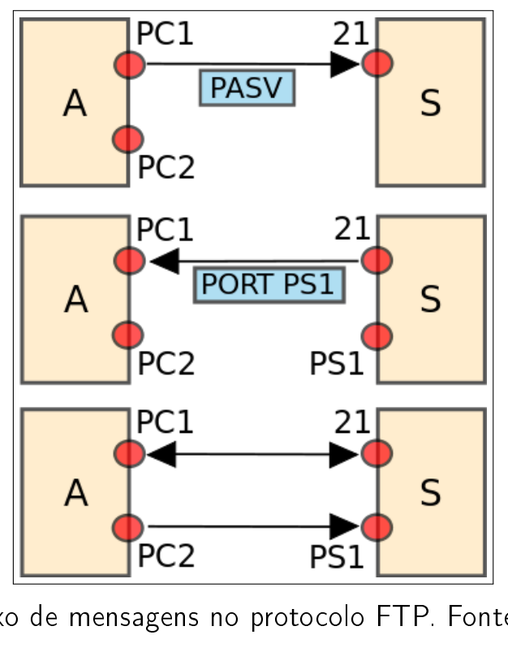


Figura: Exemplo de fluxo de mensagens no protocolo FTP. Fonte: Wikimedia Commons

Cada mensagem tem um significado exato intencionado a provocar uma resposta particular do receptor. Portanto, o protocolo define a sintaxe, semânticas, e sincronização da comunicação. Protocolos de comunicação devem ser acordados por todas as partes envolvidas.

Etherclan

A rede Etherclan é uma rede Peer-to-Peer com o objetivo de permitir que algum cliente encontre outros clientes pertinentes. Ela tem as seguintes propriedades:

- Não depende de um servidor centralizado.
- Pode ser utilizada por mais de um jogo simultaneamente.
- Capaz de buscar por membros de um jogo específico.

Para que os nós consigam conversar entre si, foi necessário definir um protocolo. Nele, foram existem as seguintes mensagems:

- ANNOUNCE_SELF – Anuncia a própria existência para o receptor.
- REQUEST_NODE_LIST – Requisita do receptor uma lista de nós da rede.
- KEEP_ALIVE – Avisa que vai enviar mais mensagens nessa mesma conexão.
- SERVICE – Comando para ser tratado usando alguma extensão definida por terceiros.

E a seguinte resposta possível:

- NODE_INFO – Informações sobre um nó da rede.

Resultados: Chat

Como prova de conceito, foi implementado um programa de conversação que utiliza a rede para enviar e receber mensagens de outros nós. As mensagens são enviadas para todos e sem nome, com o único identificador sendo o IP de origem.



Figura: Chat em funcionamento

```
Sending: 'KEEP_ALIVE ON'
Sending: 'ANNOUNCE_SELF F293B45B-1F70-496D-8AA1-DE148FA6AF81 64465 chat'
Sending: 'REQUEST_NODE_LIST'
Sending: 'KEEP_ALIVE OFF'
Received: 'NODE_INFO F293B45B-1F70-496D-8AA1-DE148FA6AF81 192.168.1.92 64465 chat'
Received: 'NODE_INFO 369463C4-00C2-47E8-8F20-0E158FF92ED4 192.168.1.92 64464 chat'
Finish
```

Figura: Log das mensagens enviadas e recebidas durante a descoberta de nós

Resultados obtidos com esse exemplo incluem:

- Funciona! É possível conseguir encontrar a rede inteira através de alguns nós.
- Problemas com NAT. Sem adotar pelo menos UPnP, torna-se impraticável o uso para encontrar nós atravéz da Internet.
- É necessário ter boas escolhas em quando iniciar novas buscas por nós, assim como por quanto tempo executar a busca.
- Nós não avisam quando eles ficam offline, então algum critério para decidir se um nó deixou de existir é importante.

Trabalhos Futuros

Durante o desenvolvimento do protocolo e principalmente da prova de conceito, vários pontos se mostraram fundamentais e que serão estudados no futuro.

- Política de quantos outros nós buscar.
- Política de quando que um nó é considerado morto e deve ser ignorado.
- Diminuir a complexidade de para quais nós devo perguntar para algo melhor que O(n).
- Desenvolvimento de uma biblioteca em C++ para facilitar a integração com outros projetos.