

Projeto de Implementação do protocolo Gnutella

Revisão do Documento 0.6

César Kallas - cesarkallas at gmx.net – RA: 02099224
Daniel Machado - daniel.fm at uol.com.br – RA: 02508224
Fábio Generoso - fhgeneroso at gmail.com - RA: 02164473
Renato Silva - renatodemarco at gmail.com – RA: 02142818

Introdução

Esse projeto visa a implementação de um servent da rede Gnutella, de acordo com a especificação do protocolo Gnutella versão 0.4, revisão 1.6 .

O servent é dividido em duas partes, uma servidor e outra cliente. Porém, as duas partes são vistas apenas como uma pelo usuário, podendo ser implementadas separadamente.

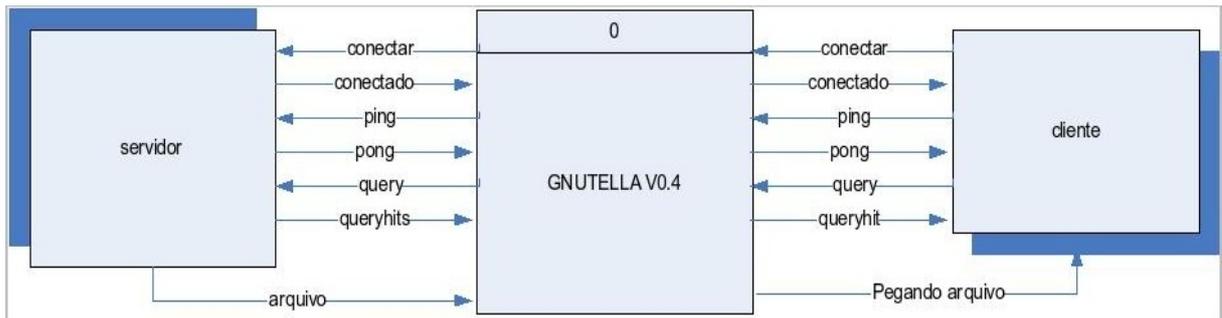
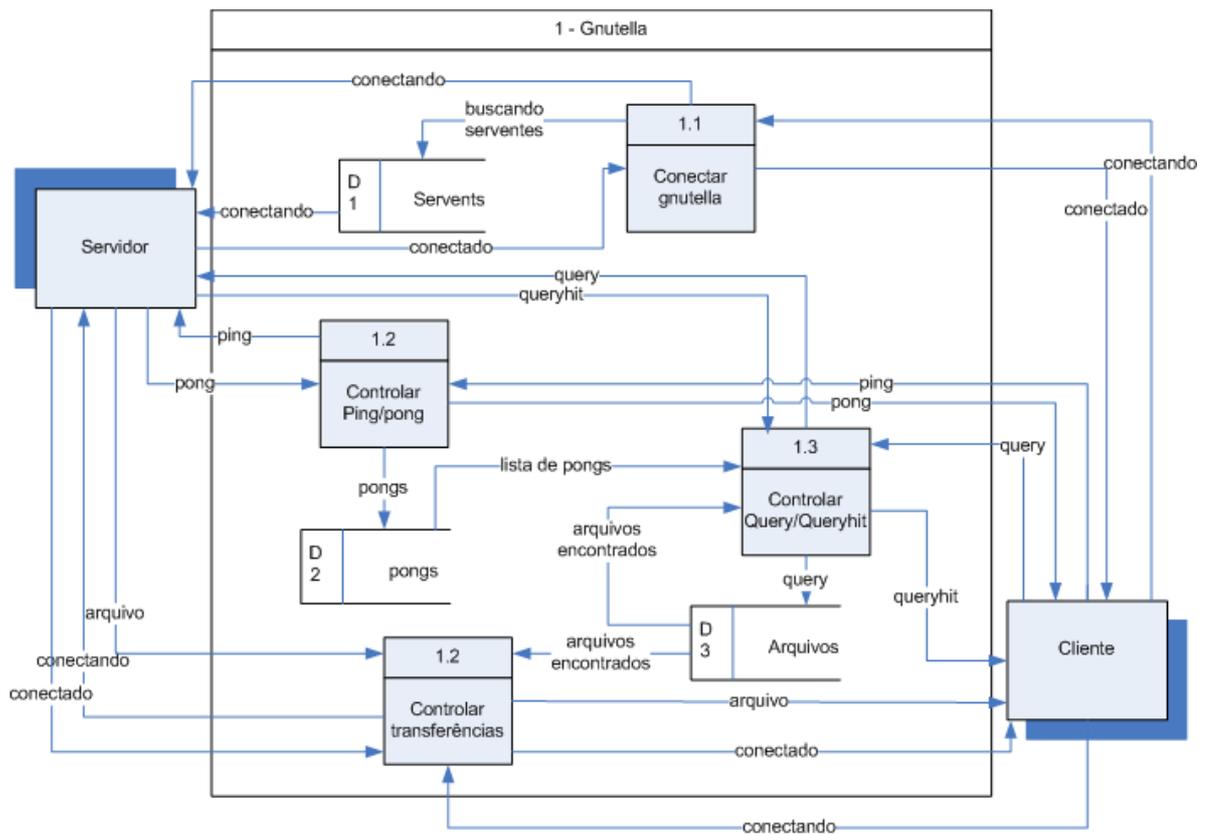


Figura 1 - DFD nível 0



DFD nível 1: Servent Gnutella

Funções

- Mensagem

Função que envia mensagem.

A mensagem pode ser enviada para um usuário ou via um socket padrão, depende do tipo passado.

Entrada: tipo da mensagem e o corpo da mensagem

Saída: status do envio da mensagem.

status função Mensagem(tipo, msg)

verifique o tipo

se tipo = USUARIO:

mostre uma mensagem na tela com msg

senão

escreva no canal de comunicação a msg

fim

retorne *status*

fim

- Conexão

É feita através de uma consulta na lista de servents.

Consulta-se o primeiro host declarado na estrutura e tenta fazer a conexão, caso não tenha sucesso, segue para o próximo host. Se todas tentativas falharem, uma mensagem é enviada para o usuário.

Entrada: lista_servents

Saída: status da conexão

função conexao(arquivo LISTA_SERVENTS)

laço: enquanto possuir host na lista_servents faça:

pegue o host no arquivo

conectar(host, porta)

se falha: volta ao inicio do laço de repetição

se fim de arquivo:

mensagem('USUARIO', "A conexão falhou, tente outros servents.")

fim

fim

status funcao conectar(host, porta)

conectar no host e na porta

mensagem(CONEXAO, host, 'GNUTELLA_VERSAO');

se 'GNUTELLA_OK':

retorne 'CONEXAO_SUCEDIDA'

se não: retorne 'FALHA_CONEXAO'

fim

- Ping

É enviado uma mensagem do tipo ping para um servent e espera-se que ele responda com pong.

Entrada: host

```

estrutura_payload função ping (host)
    BYTE ping[0:L-1]
    retorne ping
fim

```

- Pong

Resposta via mensagem com o descritor pong.

O host é o servent que enviou o ping, mas não o servent originou o ping, mas sim o servent que transmitiu por último o ping.

Entrada: porta do servent, host, número de arquivos no servent e tamanho total desses arquivos.

```

estrutura_payload função pong()
    BYTE pong[L-1]
    pong[0:1] = PORTA_ENTRADA
    pong[2:5] = getLocalIpAddress()
    pong[6:9] = ArqCompartilhados(N_ARQUIVOS, NULL)
    pong[10:13] = ArqCompartilhados(TAM_ARQUIVOS, NULL)
    retorne pong
fim

```

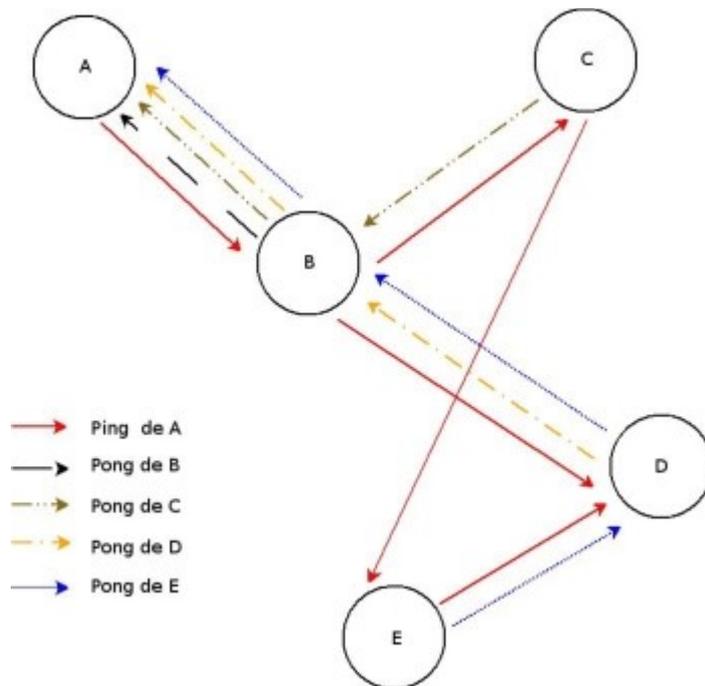


Figura 3 - Percursso de um PING / PONG

- Query

Função de busca de arquivos no servents.

```

estrutura_payload função query(string)
    BYTE query[L-1]
    query[0:1] = QUERY_LIVRE | QUERY_MIN | QUERY_MED
    inteiro n = tamanho(string)

```

```

query[2:n] = string
query[n+1] = NULL
retorne query

```

fim

- QueryHits

Função que retorna o número de arquivos encontrados de acordo com uma query passada.

```

estrutura_payload função queryhits(string)
BYTE queryhits[L-1]
queryhits[1:2] = PORTA_ENTRADA
queryhits[3:6] = getlocalipaddress()
queryhits[7:10] = getSpeed()
queryhits[11:N] = ArqCompartilhados(PROC_ARQUIVOS, string)
retorne estrutura_payload

```

fim

- GeraDescritor

Função que encapsula o payload no descritor.

Array de bytes:

Descritor ID	Payload	TTL	Hops	Tamanho do Payload
--------------	---------	-----	------	--------------------

Entrada:

Payload - PING / PONG / QUERY / QUERYHITS / BYE
Estrutura correspondente ao descritor payload

```

descricao função GeraDescritor(payload, estrutura_payload)
BYTE descritor[n]
descricao[0:15] = GeraDescriptorID()
descricao[16] = payload
descricao[17] = TTL
descricao[18] = 0
inteiro t = tamanho(estrutura_payload)
descricao[19-22] = t
descricao[23:t] = estrutura_payload
retorne descritor

```

fim

- GeraDescriptorID

Função que gera um número de 16 bytes (128 bits) aleatório (e possivelmente único).

```

numero função GeraDescriptorID()
inteiro n = gera_id_randomico(128 bits)
retorne n

```

fim

- ArqCompartilhados

Retorna o número de arquivos compartilhados ou o tamanho total desses arquivos. Entrada:

String: nome de um arquivo

Opção: N_ARQUIVOS (número de arquivos)

TAM_ARQUIVOS (tamanho total dos arquivos)

numero ArqCompartilhados(opcao, string)

inteiro n = some o número de arquivos no diretório

verifique opcao

caso TAM_ARQUIVOS:

laço: enquanto n != 0

string arquivo = prox_arq_diretório(DIRETÓRIO)

tam = tam + tamanho(arquivo)

n - -

retorne tam

caso N_ARQUIVOS:

retorne n

caso PROC_ARQUIVOS:

estrutura dados_arquivo (inteiro index,

inteiro tamanho, string nome)

procure_no_disco(string, arquivo)

retorne dados_arquivo

fim

fim

- Principal

Função principal do programa.

função principal()

conexao(LISTA_SERVENTS)

nova_linha_execucao(servidor(PORTA_ENTRADA))

...

servidor:

laço:

BYTE req[] = escute_porta(PORTA_ENTRADA)

se req[16] = PING

nova_linha_execucao(

GeraDescriptorID(PONG, pong()))

se req[16] = QUERY

nova_linha_execucao(

GeraDescriptorID(QUERYHITS,

queryhits(req[2:N])))

...

...

cliente

Mensagem(PING, GeraDescriptorID(PING, ping()))

Mensagem(QUERY, GeraDescriptorID(QUERY, query(string)))

...

fim

Estrutura de dados

- Constantes

Contém a definição de todas as constantes do protocolo, declarados de forma estática em um função do programa separada ou na função principal.

string	GNUTELLA	“GNUTELLA CONNECT/0.4\n\n”
string	GNUTELLA_OK	“GNUTELLA OK\n\n”
string	DIRETORIO	“/tmp/gnutella/arquivos”
inteiro	CONEXAO	0xa0
inteiro	CONEXAO_SUCEDIDA	0xa1
inteiro	FALHA_CONEXAO	0xa2
inteiro	USUARIO	0xa3
inteiro	N_ARQUIVOS	0xa4
inteiro	TAM_ARQUIVOS	0xa5
inteiro	PROC_ARQUIVOS	0xa6
inteiro	PING	0x00
inteiro	PONG	0x01
inteiro	QUERY	0x80
inteiro	QUERYHITS	0x81
inteiro	BYE	0x02
inteiro	TTL	10
inteiro	QUERY_LIVRE	0x00
inteiro	QUERY_MIN	0x01
inteiro	QUERY_MED	0x02
inteiro	NULL	0x00
string	LISTA_SERVENTS	“servents.ini”

- Variáveis

inteiro	PORTA_ENTRADA	2000 (padrão)
---------	---------------	---------------

- Lista de servents: lista_servents

Arquivo texto com hosts que já estão conectados a rede, lista obtida através de sites na internet.

Estrutura do arquivo:

<nome do host>:<porta>

gnet.ath.cx:6346

gnet2.ath.cx:6346

gnet3.ath.cx:6346

gnet4.ath.cx:6346

gnet5.ath.cx:6346

gnotella.fileflash.com:6346

gnutellahosts.com:6346

- Diretório de arquivos

Diretório em disco que contém os arquivos a serem compartilhados na rede.

- Lista de arquivos sendo baixados

Estrutura de dados com a lista de arquivos sendo baixos pelo cliente do servent.

Bibliografia

- <http://rfc-gnutella.sourceforge.net/>
- <http://www.gnutelliums.com/>
- <http://gnutella.da.ru/> (lista de hosts.)