



UNIVERSIDADE DO MINHO

ARQUITETURAS EMERGENTES DE REDE

IMPLEMENTAÇÃO DE UM SERVIÇO DE DIFUSÃO DE  
INFORMAÇÃO DE TRÂNSITO NUMA REDE VEICULAR  
(VANET DE DADOS NOMEADOS)

João Freitas A74814

João Mendes A71862

May 31, 2020

# Índice

<b>1</b>	<b>Introdução</b>	<b>iii</b>
<b>2</b>	<b>Implementação e Resultados</b>	<b>iv</b>
2.1	Fase 1 . . . . .	iv
2.1.1	Protocolo HELLO . . . . .	iv
2.1.2	Protocolo ROUTE_REQUEST . . . . .	vi
2.2	Fase 2 . . . . .	viii
2.2.1	Submissão de informação . . . . .	viii
2.2.2	Consulta de informação . . . . .	ix
<b>3</b>	<b>Conclusão</b>	<b>xi</b>

# Índice de Imagens

2.1	Cenário core a correr protocolo HELLO e ROUTE_REQUEST . . . . .	iv
2.2	Output protocolo HELLO nó 1 . . . . .	v
2.3	Output protocolo HELLO nó 2 . . . . .	v
2.4	Output protocolo HELLO nó 3 . . . . .	v
2.5	Output protocolo HELLO nó 5 . . . . .	v
2.6	Output protocolo HELLO nó 6 . . . . .	v
2.7	Output protocolo ROUTE_REQUEST nó 1 . . . . .	vi
2.8	Output protocolo ROUTE_REQUEST nó 2 . . . . .	vi
2.9	Output protocolo ROUTE_REQUEST nó 3 . . . . .	vii
2.10	Submissão info nó1 . . . . .	viii
2.11	Submissão info nó2 . . . . .	ix
2.12	Submissão info nó3 . . . . .	ix
2.13	Consulta info nó1 . . . . .	x
2.14	Consulta info nó2 . . . . .	x
2.15	Consulta info nó3 . . . . .	x

# Chapter 1

## Introdução

Neste relatório iremos expor o processo de desenvolvimento do trabalho prático da Unidade Curricular de Arquiteturas Emergentes de Rede. Este trabalho teve como objetivo aprofundar conhecimentos de como as redes Adhoc funcionam, nas quais, ao contrário das redes infraestruturadas, não há separação clara entre nós terminais e nós encaminhadores de tráfego, já que cada um dos nós tem de desempenhar tanto as funções de receber tráfego e encaminhar tráfego.

Neste caso em específico falamos de redes VANET(Vehicular Adhoc Network), logo a topologia terá que ser muito dinâmica já que os nós podem-se mover a grandes velocidades e perder ligações muito frequentemente logo é necessária a implementação de algoritmos de comunicação eficientes para o encaminhamento de tráfego neste tipo de redes. As arquiteturas Adhoc juntamente com as NDNs(Named Data Networks) possibilitam essas redes VANET, já que nas redes NDNs os dados não são endereçados a um nó particular mas sim a um nome, sendo o nome, neste caso, as informações de trânsito presentes nos nós.

Para o teste da implementação dos algoritmos criados foi criada uma rede core emulada com 5 nós. Nesta rede serão testados os protocolos HELLO e ROUTE\_REQUEST e também a implementação do serviço de difusão de informações de trânsito nesta rede.

## Chapter 2

# Implementação e Resultados

Serão abordadas aqui as implementações da Fase 1 e Fase 2 deste trabalho.

## 2.1 Fase 1

### 2.1.1 Protocolo HELLO

Para a implementação do protocolo HELLO foram criadas duas estruturas de dados principais de forma a facilitar os protocolos subsequentes a implementar. Essas estruturas são a estrutura **Vizinhos** e a estrutura **Caminhos**. Na estrutura **Vizinhos** temos os vizinhos diretos do nó em questão e os vizinhos desses vizinhos diretos, isto é, apenas os que estão a 2 saltos de distância do nó original sendo que estes sabem qual é o vizinho pelo qual são conhecidos. A estrutura **Caminhos** é algo semelhante, mas temos agora o caminho para esse vizinho que o nó original conhece.

Olhando para as figuras em baixo, com o output do programa, será mais fácil a compreensão.

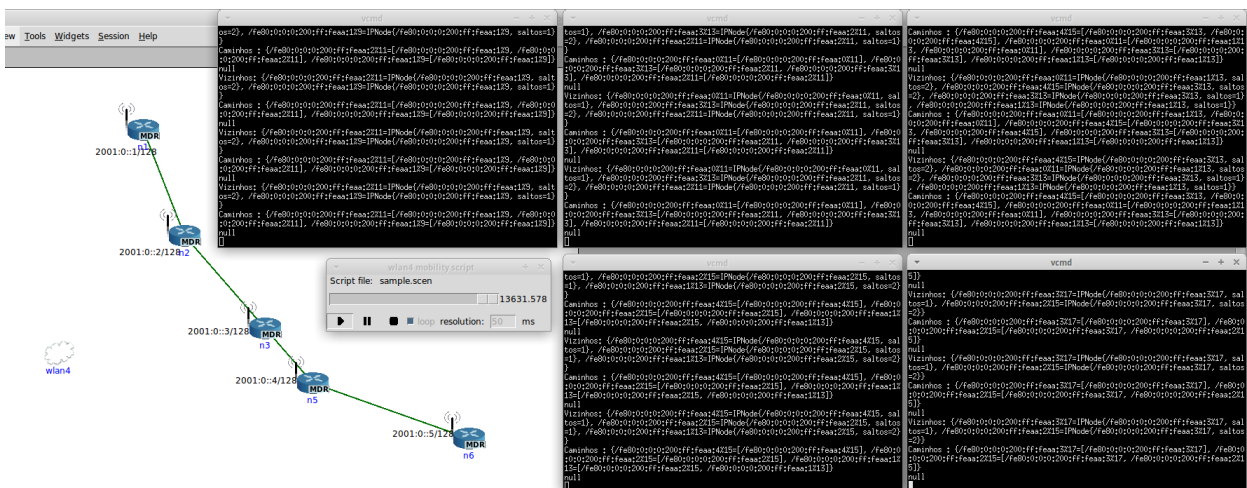


Figure 2.1: Cenário core a correr protocolo HELLO e ROUTE\_REQUEST

```
Vizinhos: {/fe80:0:0:0:200:ff:feaa:2%11=IPNode{/fe80:0:0:0:200:ff:feaa:1%9, saltos=2}, /fe80:0:0:0:200:ff:feaa:1%9=IPNode{/fe80:0:0:0:200:ff:feaa:1%9, saltos=1}}
Caminhos : {/fe80:0:0:0:200:ff:feaa:2%11=[/fe80:0:0:0:200:ff:feaa:1%9, /fe80:0:0:0:200:ff:feaa:2%11], /fe80:0:0:0:200:ff:feaa:1%9=[/fe80:0:0:0:200:ff:feaa:1%9]}
null
^Croot@n1:/home/core/NetBeansProjects/AER/src#
```

Figure 2.2: Output protocolo HELLO nó 1

```
Vizinhos: {/fe80:0:0:0:200:ff:feaa:0%11=IPNode{/fe80:0:0:0:200:ff:feaa:0%11, saltos=1}, /fe80:0:0:0:200:ff:feaa:3%13=IPNode{/fe80:0:0:0:200:ff:feaa:2%11, saltos=2}, /fe80:0:0:0:200:ff:feaa:2%11=IPNode{/fe80:0:0:0:200:ff:feaa:2%11, saltos=1}}
Caminhos : {/fe80:0:0:0:200:ff:feaa:0%11=[/fe80:0:0:0:200:ff:feaa:0%11], /fe80:0:0:0:200:ff:feaa:3%13=[/fe80:0:0:0:200:ff:feaa:2%11, /fe80:0:0:0:200:ff:feaa:3%13], /fe80:0:0:0:200:ff:feaa:2%11=[/fe80:0:0:0:200:ff:feaa:2%11]}
null
^Croot@n2:/home/core/NetBeansProjects/AER/src#
```

Figure 2.3: Output protocolo HELLO nó 2

```
Vizinhos: {/fe80:0:0:0:200:ff:feaa:4%15=IPNode{/fe80:0:0:0:200:ff:feaa:3%13, saltos=2}, /fe80:0:0:0:200:ff:feaa:0%11=IPNode{/fe80:0:0:0:200:ff:feaa:1%13, saltos=2}, /fe80:0:0:0:200:ff:feaa:3%13=IPNode{/fe80:0:0:0:200:ff:feaa:3%13, saltos=1}, /fe80:0:0:0:200:ff:feaa:1%13=IPNode{/fe80:0:0:0:200:ff:feaa:1%13, saltos=1}}
Caminhos : {/fe80:0:0:0:200:ff:feaa:4%15=[/fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4%15], /fe80:0:0:0:200:ff:feaa:0%11=[/fe80:0:0:0:200:ff:feaa:1%13, /fe80:0:0:0:200:ff:feaa:0%11], /fe80:0:0:0:200:ff:feaa:3%13=[/fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:1%13]}
null
^Croot@n3:/home/core/NetBeansProjects/AER/src#
```

Figure 2.4: Output protocolo HELLO nó 3

```
Vizinhos: {/fe80:0:0:0:200:ff:feaa:4%15=IPNode{/fe80:0:0:0:200:ff:feaa:4%15, saltos=1}, /fe80:0:0:0:200:ff:feaa:2%15=IPNode{/fe80:0:0:0:200:ff:feaa:2%15, saltos=1}, /fe80:0:0:0:200:ff:feaa:1%13=IPNode{/fe80:0:0:0:200:ff:feaa:2%15, saltos=2}}
Caminhos : {/fe80:0:0:0:200:ff:feaa:4%15=[/fe80:0:0:0:200:ff:feaa:4%15], /fe80:0:0:0:200:ff:feaa:2%15=[/fe80:0:0:0:200:ff:feaa:2%15], /fe80:0:0:0:200:ff:feaa:1%13=[/fe80:0:0:0:200:ff:feaa:2%15, /fe80:0:0:0:200:ff:feaa:1%13]}
null
^Croot@n5:/home/core/NetBeansProjects/AER/src#
```

Figure 2.5: Output protocolo HELLO nó 5

```
Vizinhos: {/fe80:0:0:0:200:ff:feaa:3%17=IPNode{/fe80:0:0:0:200:ff:feaa:3%17, saltos=1}, /fe80:0:0:0:200:ff:feaa:2%15=IPNode{/fe80:0:0:0:200:ff:feaa:3%17, saltos=2}}
Caminhos : {/fe80:0:0:0:200:ff:feaa:3%17=[/fe80:0:0:0:200:ff:feaa:3%17], /fe80:0:0:0:200:ff:feaa:2%15=[/fe80:0:0:0:200:ff:feaa:3%17, /fe80:0:0:0:200:ff:feaa:2%15]}
null
^Croot@n6:/home/core/NetBeansProjects/AER/src#
```

Figure 2.6: Output protocolo HELLO nó 6

### 2.1.2 Protocolo ROUTE\_REQUEST

Este protocolo pretende obter a rota para nós fora da vizinhança  $V=2$ , sendo para isso necessário mandar pacotes do tipo `route_request` em flooding, isto é, para todos os nós ligados ao nó original, até que um destes conheça o caminho para o nó destino, ou seja o nó destino, escolhido pelo utilizador. Caso seja, deverá ser devolvido esse caminho até ao nó original com o tipo de pacote `route_reply`.

Segue-se o output quando é feito o pedido de `route_request` do nó 1 para o nó 6. De notar os caminhos atualizados para o nó com IP `/fe80:0:0:0:200:ff:feaa:4` (nó 6), tanto no nó 1 como no nó 2.

```
Escreva a opção:
1-Indicar que é servidor de difusão
2-Indicar que é cliente de difusão
3-Pedido de rota
4-Pedido de consulta de informação
5-Pedido de submissão de informação
/fe80:0:0:0:200:ff:feaa:4
Sent RouteRequest:
RouteRequest{ipdestino=/fe80:0:0:0:200:ff:feaa:4, radius=5, caminho=[]}
Reached the original node, no need to send reply
Reached the original node, no need to send reply
Reached the original node, no need to send reply
Caminhos : {/fe80:0:0:0:200:ff:feaa:4=[/fe80:0:0:0:200:ff:feaa:1%11, /fe80:0:0:0:200:ff:feaa:2%13, /fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4], /fe80:0:0:0:200:ff:feaa:2%11=[/fe80:0:0:0:200:ff:feaa:1%9, /fe80:0:0:0:200:ff:feaa:2%11], /fe80:0:0:0:200:ff:feaa:1%9=[/fe80:0:0:0:200:ff:feaa:1%9]}
Caminhos : {/fe80:0:0:0:200:ff:feaa:4=[/fe80:0:0:0:200:ff:feaa:1%11, /fe80:0:0:0:200:ff:feaa:2%13, /fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4], /fe80:0:0:0:200:ff:feaa:2%11=[/fe80:0:0:0:200:ff:feaa:1%9, /fe80:0:0:0:200:ff:feaa:2%11], /fe80:0:0:0:200:ff:feaa:1%9=[/fe80:0:0:0:200:ff:feaa:1%9]}
^Croot@n1:/home/core/NetBeansProjects/AER/src#
```

Figure 2.7: Output protocolo ROUTE\_REQUEST nó 1

```
Received Request:
RouteRequest{ipdestino=/fe80:0:0:0:200:ff:feaa:4, radius=4, caminho=[]}
Caminho para routereply[/fe80:0:0:0:200:ff:feaa:1%11]
Sent Request:
RouteRequest{ipdestino=/fe80:0:0:0:200:ff:feaa:4, radius=4, caminho=[/fe80:0:0:0:200:ff:feaa:1%11]}
Sent Reply: RouteReply{caminho=[/fe80:0:0:0:200:ff:feaa:1%11, /fe80:0:0:0:200:ff:feaa:2%13, /fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4], contador=1, ipdestino=/fe80:0:0:0:200:ff:feaa:4, iporigem=/fe80:0:0:0:200:ff:feaa:0%9, radius=4}
Sent Reply: RouteReply{caminho=[/fe80:0:0:0:200:ff:feaa:1%11, /fe80:0:0:0:200:ff:feaa:2%13, /fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4], contador=1, ipdestino=/fe80:0:0:0:200:ff:feaa:4, iporigem=/fe80:0:0:0:200:ff:feaa:0%9, radius=2}
Sent Reply: RouteReply{caminho=[/fe80:0:0:0:200:ff:feaa:1%11, /fe80:0:0:0:200:ff:feaa:2%13, /fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4], contador=1, ipdestino=/fe80:0:0:0:200:ff:feaa:4, iporigem=/fe80:0:0:0:200:ff:feaa:0%9, radius=2}
Caminhos : {/fe80:0:0:0:200:ff:feaa:4=[/fe80:0:0:0:200:ff:feaa:2%13, /fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4], /fe80:0:0:0:200:ff:feaa:0%11=[/fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:2%11], /fe80:0:0:0:200:ff:feaa:3%13=[/fe80:0:0:0:200:ff:feaa:2%11, /fe80:0:0:0:200:ff:feaa:3%13], /fe80:0:0:0:200:ff:feaa:2%11=[/fe80:0:0:0:200:ff:feaa:2%11]}
```

Figure 2.8: Output protocolo ROUTE\_REQUEST nó 2

```

Received Request:
RouteRequest{ipdestino=/fe80:0:0:0:200:ff:feaa:4, radius=3, caminho=[/fe80:0:0:0:200:ff:feaa:1%11]}
Caminho para routereply[/fe80:0:0:0:200:ff:feaa:1%11, /fe80:0:0:0:200:ff:feaa:2%13]
Sent Reply: RouteReply{caminho=[/fe80:0:0:0:200:ff:feaa:1%11, /fe80:0:0:0:200:ff:feaa:2%13, /fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4], contador=1, ipdestino=/fe80:0:0:0:200:ff:feaa:4, iporigem=/fe80:0:0:0:200:ff:feaa:0%29, radius=5}
Sent Reply: RouteReply{caminho=[/fe80:0:0:0:200:ff:feaa:1%11, /fe80:0:0:0:200:ff:feaa:2%13, /fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4], contador=1, ipdestino=/fe80:0:0:0:200:ff:feaa:4, iporigem=/fe80:0:0:0:200:ff:feaa:0%29, radius=3}
Sent Reply: RouteReply{caminho=[/fe80:0:0:0:200:ff:feaa:1%11, /fe80:0:0:0:200:ff:feaa:2%13, /fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4], contador=1, ipdestino=/fe80:0:0:0:200:ff:feaa:4, iporigem=/fe80:0:0:0:200:ff:feaa:0%29, radius=3}
Sent Reply: RouteReply{caminho=[/fe80:0:0:0:200:ff:feaa:1%11, /fe80:0:0:0:200:ff:feaa:2%13, /fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4], contador=1, ipdestino=/fe80:0:0:0:200:ff:feaa:4, iporigem=/fe80:0:0:0:200:ff:feaa:0%29, radius=1}
Sent Reply: RouteReply{caminho=[/fe80:0:0:0:200:ff:feaa:1%11, /fe80:0:0:0:200:ff:feaa:2%13, /fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4], contador=1, ipdestino=/fe80:0:0:0:200:ff:feaa:4, iporigem=/fe80:0:0:0:200:ff:feaa:0%29, radius=1}
Sent Reply: RouteReply{caminho=[/fe80:0:0:0:200:ff:feaa:1%11, /fe80:0:0:0:200:ff:feaa:2%13, /fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4], contador=1, ipdestino=/fe80:0:0:0:200:ff:feaa:4, iporigem=/fe80:0:0:0:200:ff:feaa:0%29, radius=1}
Sent Reply: RouteReply{caminho=[/fe80:0:0:0:200:ff:feaa:1%11, /fe80:0:0:0:200:ff:feaa:2%13, /fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4], contador=1, ipdestino=/fe80:0:0:0:200:ff:feaa:4, iporigem=/fe80:0:0:0:200:ff:feaa:0%29, radius=1}
Sent Reply: RouteReply{caminho=[/fe80:0:0:0:200:ff:feaa:1%11, /fe80:0:0:0:200:ff:feaa:2%13, /fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4], contador=1, ipdestino=/fe80:0:0:0:200:ff:feaa:4, iporigem=/fe80:0:0:0:200:ff:feaa:0%29, radius=1}
Caminhos : {/fe80:0:0:0:200:ff:feaa:4%15=[/fe80:0:0:0:200:ff:feaa:3%13, /fe80:0:0:0:200:ff:feaa:4%15], /fe80:0:0:0:200:ff:feaa:0%11=[/fe80:0:0:0:200:ff:feaa:1%13, /fe80:0:0:0:200:ff:feaa:0%11], /fe80:0:0:0:200:ff:feaa:3%13=[/fe80:0:0:0:200:ff:feaa:3%13], /fe80:0:0:0:200:ff:feaa:1%13=[/fe80:0:0:0:200:ff:feaa:1%13]}

```

Figure 2.9: Output protocolo ROUTE\_REQUEST nó 3



## 2.2 Fase 2

Nesta fase é implementada a difusão das informações de trânsito pela rede, sendo esta informação uma String guardada em cada nó. O utilizador de cada nó tem a opção de se tornar cliente ou servidor sendo que o cliente pode submeter e consultar informações de servidores. Infelizmente não conseguimos obter ligação a nós a mais de dois saltos pois ocorria *IOException : rede inacessível* quando tentava conectar a esses nós. Apesar disso conseguimos conectar do nó1 ao nó3(dois saltos de distância).

### 2.2.1 Submissão de informação

Output de submissão de informação feita do nó1(/fe80:0:0:0:200:ff:feaa:0%9) ao nó3(/fe80:0:0:0:200:ff:feaa:2) com info= *ola*.

```
Escreva IP destino:
null
2
Escreva info:
null
olanull

Efetuado pedido de consulta.
Escreva a opção:
1-Indicar que é servidor de difusão
2-Indicar que é cliente de difusão
3-Pedido de rota
4-Pedido de consulta de informação
5-Pedido de submissão de informação
Connection from Socket[addr=/fe80:0:0:0:200:ff:feaa:0%9,port=40601,localport=999
9]
Connected
Get info message.
Received message MessageInfo{iporigem=/fe80:0:0:0:200:ff:feaa:0%9, ipdestino=/fe
80:0:0:0:200:ff:feaa:2, info=ola, opt=5, retorno=false}
Awaiting connections...
Closing socket
null
^Croot@n1:/home/core/NetBeansProjects/AER/src#
```

Figure 2.10: Submissão info nó1

```

^Croot@n2:/home/core/NetBeansProjects/AER/src# java aer.MsgBroadcaster
Awaiting connections...
ConsoleInputReadTask run() called.
Escreva a opção:
1-Indicar que é servidor de difusão
2-Indicar que é cliente de difusão
3-Pedido de rota
4-Pedido de consulta de informação
5-Pedido de submissão de informação
null
null
null
null
null
null
null
null
null
Received info msg:ola
/fe80:0:0:0:200:ff:feaa:2%11
null
null
null
^Croot@n2:/home/core/NetBeansProjects/AER/src# 

```

Figure 2.11: Submissão info nó2

```

1
Nó servidor
Escreva a opção:
1-Indicar que é servidor de difusão
2-Indicar que é cliente de difusão
3-Pedido de rota
4-Pedido de consulta de informação
5-Pedido de submissão de informação
null
null
null
null
Received info msg:ola
Connection from Socket[addr=/fe80:0:0:0:200:ff:feaa:2%13,port=47037,localport=99
99]
Received message MessageInfo{iporigem=/fe80:0:0:0:200:ff:feaa:0%9, ipdestino=/fe
80:0:0:0:200:ff:feaa:2, info=ola, opt=5, retorno=false}
Mensagem recebida pelo IPdestinoMessageInfo{iporigem=/fe80:0:0:0:200:ff:feaa:0%9
, ipdestino=/fe80:0:0:0:200:ff:feaa:2, info=ola, opt=5, retorno=false}
Awaiting connections...
ola
^Croot@n3:/home/core/NetBeansProjects/AER/src# 

```

Figure 2.12: Submissão info nó3

### 2.2.2 Consulta de informação

Output de consulta de informação feita do nó1(/fe80:0:0:0:200:ff:feaa:0%9) ao nó3(/fe80:0:0:0:200:ff:feaa:2) com info= *ola*.

O output apenas aparece no nó1 de forma inconsistente.

```

Received info msg:ola
Connection from Socket[addr=/fe80:0:0:0:200:ff:feaa:0%9,port=40652,localport=999]
Received message MessageInfo{iporigem=/fe80:0:0:0:200:ff:feaa:2%13, ipdestino=/fe80:0:0:0:200:ff:feaa:0%9, info=ola, opt=4, retorno=true}
Awaiting connections...
null
^Croot@n1:/home/core/NetBeansProjects/AER/src# █

```

Figure 2.13: Consulta info nó1

```

Received info msg:null
/fe80:0:0:0:200:ff:feaa:2%11
null
Received info msg:ola
/fe80:0:0:0:200:ff:feaa:0%11
null
^Croot@n2:/home/core/NetBeansProjects/AER/src# █

```

Figure 2.14: Consulta info nó2

```

Connection from Socket[addr=/fe80:0:0:0:200:ff:feaa:2%13,port=47086,localport=999]
Received message MessageInfo{iporigem=/fe80:0:0:0:200:ff:feaa:0%9, ipdestino=/fe80:0:0:0:200:ff:feaa:2, info=null, opt=4, retorno=false}
Sent server msg:MessageInfo{iporigem=/fe80:0:0:0:200:ff:feaa:2%13, ipdestino=/fe80:0:0:0:200:ff:feaa:0%9, info=ola, opt=4, retorno=true}
Mensagem recebida pelo IPdestinoola
Awaiting connections...
ola
^Croot@n3:/home/core/NetBeansProjects/AER/src# █

```

Figure 2.15: Consulta info nó3

## Chapter 3

# Conclusão

Infelizmente não foi possível a conclusão deste trabalho prático, sendo que não conseguimos implementar a Fase 3 relacionada com as NDNs para obter as informações de trânsitos através de nomes e não através de IPs que o utilizador requisita. Para além disso a Fase 2 não está a funcionar de forma consistente na consulta de informação e pelo facto de não conseguir comunicar com nós fora da vizinhança  $V=2$ .

Apesar desse pontos negativos pensamos que aprofundamos bastante os conhecimentos de programação distribuída na implementação de algoritmos emergentes, já que conseguimos implementar com relativo sucesso a fase 1 e 2 deste trabalho.