



# **TP3: Serviço de Resolução de Nomes (DNS)**

Comunicações por Computador

Grupo 1 – PL5



Carla Cruz A80564



Ana Ribeiro A82474



Jéssica Lemos A82061

## QUESTÕES E RESPOSTAS

### PARTE I

- a. Qual o conteúdo do ficheiro `/etc/resolv.conf` e para que serve essa informação?

```
# This file is not consulted for DNS hostname resolution, address
# resolution, or the DNS query routing mechanism used by most
# processes on this system.
#
# To view the DNS configuration used by this system, use:
#   scutil --dns
#
# SEE ALSO
#   dns-sd(1), scutil(8)
#
# This file is automatically generated.
#
domain eduroam.uminho.pt
nameserver 193.137.16.65
nameserver 193.137.16.145
nameserver 193.137.16.75
```

Figura 1 - Conteúdo do ficheiro `/etc/resolv.conf`

O ficheiro `/etc/resolv.conf` possui informação acerca das configurações default do sistema, nomeadamente os servidores de nome e o domínio em que nos encontramos, sendo estes utilizados para aceder ao serviço de DNS.

- b. Os servidores `www.google.pt` e `www.google.com` têm endereços IPv6? Se sim, quais?

Para obter os endereços IPv6 dos servidores, realizamos uma query do tipo AAAA.

```
core@XubunCORE:~$ nslookup
> set q=AAAA
> www.google.pt
Server:      193.137.16.65
Address:     193.137.16.65#53

Non-authoritative answer:
www.google.pt has AAAA address 2a00:1450:4003:80a::2003

Authoritative answers can be found from:
pt      nameserver = sns-pb.isc.org.
pt      nameserver = g.dns.pt.
pt      nameserver = d.dns.pt.
pt      nameserver = e.dns.pt.
pt      nameserver = ns.dns.br.
pt      nameserver = a.dns.pt.
pt      nameserver = b.dns.pt.
pt      nameserver = f.dns.pt.
pt      nameserver = ns2.nic.fr.
pt      nameserver = c.dns.pt.
a.dns.pt internet address = 185.39.208.1
b.dns.pt internet address = 194.0.25.23
c.dns.pt internet address = 204.61.216.105
d.dns.pt internet address = 185.39.210.1
e.dns.pt internet address = 193.136.192.64
f.dns.pt internet address = 162.88.45.1
g.dns.pt internet address = 193.136.2.226
ns.dns.br internet address = 200.160.0.5
ns2.nic.fr internet address = 192.93.0.4
sns-pb.isc.org internet address = 192.5.4.1
a.dns.pt has AAAA address 2a04:6d80::1
b.dns.pt has AAAA address 2001:678:20::23
c.dns.pt has AAAA address 2001:500:14:6105:ad::1
```

Figura 2 – Endereço IPv6 do `www.google.pt`

O servidor `www.google.pt.` tem o endereço IPv6 `2a00:1450:4003:80a::2003`, como é possível verificar na Figura 2.

```
> www.google.com
Server:      193.137.16.65
Address:     193.137.16.65#53

Non-authoritative answer:
www.google.com has AAAA address 2a00:1450:4003:80a::2004

Authoritative answers can be found from:
google.com    nameserver = ns4.google.com.
google.com    nameserver = ns2.google.com.
google.com    nameserver = ns3.google.com.
google.com    nameserver = ns1.google.com.
ns2.google.com internet address = 216.239.34.10
ns1.google.com internet address = 216.239.32.10
ns3.google.com internet address = 216.239.36.10
ns4.google.com internet address = 216.239.38.10
ns2.google.com has AAAA address 2001:4860:4802:34::a
ns1.google.com has AAAA address 2001:4860:4802:32::a
ns3.google.com has AAAA address 2001:4860:4802:36::a
ns4.google.com has AAAA address 2001:4860:4802:38::a
```

Figura 3 - Endereço IPv6 do `www.google.com`

Na Figura 3 observamos que o servidor `www.google.com.` tem o endereço IPv6 `2a00:1450:4003:80a::2004`.

**c. Quais os servidores de nomes definidos para os domínios: “ccg.pt.”, “pt.” e “.”?**

```
> set q=NS
> ccg.pt.
Server:      193.137.16.65
Address:     193.137.16.65#53

Non-authoritative answer:
ccg.pt nameserver = ns3.ccg.pt.
ccg.pt nameserver = ns1.ccg.pt.

Authoritative answers can be found from:
ns3.ccg.pt    internet address = 193.136.11.203
ns1.ccg.pt    internet address = 193.136.11.201
```

Figura 4 - Servidores de nome para o domínio “ccg.pt.”

Os servidores de nome definidos para “ccg.pt” são `ns3.ccg.pt` e `ns1.ccg.pt`, como pode ser observado na Figura 4.

```

> pt.
Server:      193.137.16.65
Address:     193.137.16.65#53

Non-authoritative answer:
pt      nameserver = b.dns.pt.
pt      nameserver = e.dns.pt.
pt      nameserver = ns.dns.br.
pt      nameserver = d.dns.pt.
pt      nameserver = c.dns.pt.
pt      nameserver = f.dns.pt.
pt      nameserver = sns-pb.isc.org.
pt      nameserver = ns2.nic.fr.
pt      nameserver = g.dns.pt.
pt      nameserver = a.dns.pt.

Authoritative answers can be found from:
a.dns.pt      internet address = 185.39.208.1
b.dns.pt      internet address = 194.0.25.23
c.dns.pt      internet address = 204.61.216.105
d.dns.pt      internet address = 185.39.210.1
e.dns.pt      internet address = 193.136.192.64
f.dns.pt      internet address = 162.88.45.1
g.dns.pt      internet address = 193.136.2.226
ns.dns.br     internet address = 200.160.0.5
ns2.nic.fr    internet address = 192.93.0.4
sns-pb.isc.org internet address = 192.5.4.1
a.dns.pt      has AAAA address 2a04:6d80::1
b.dns.pt      has AAAA address 2001:678:20::23
c.dns.pt      has AAAA address 2001:500:14:6105:ad::1
d.dns.pt      has AAAA address 2a04:6d82::1
e.dns.pt      has AAAA address 2001:690:a00:4001::64

```

Figura 5 - Servidores de nome para o domínio "pt."

Os servidores de nome definidos para "pt." são todos os nameservers apresentados na figura em cima, que neste caso são dez.

```

> .
Server:      193.137.16.65
Address:     193.137.16.65#53

Non-authoritative answer:
.      nameserver = i.root-servers.net.
.      nameserver = j.root-servers.net.
.      nameserver = h.root-servers.net.
.      nameserver = f.root-servers.net.
.      nameserver = b.root-servers.net.
.      nameserver = k.root-servers.net.
.      nameserver = e.root-servers.net.
.      nameserver = g.root-servers.net.
.      nameserver = l.root-servers.net.
.      nameserver = d.root-servers.net.
.      nameserver = c.root-servers.net.
.      nameserver = a.root-servers.net.
.      nameserver = m.root-servers.net.

Authoritative answers can be found from:
l.root-servers.net      internet address = 199.7.83.42
b.root-servers.net      internet address = 199.9.14.201
h.root-servers.net      internet address = 198.97.190.53
j.root-servers.net      internet address = 192.58.128.30
m.root-servers.net      internet address = 202.12.27.33
k.root-servers.net      internet address = 193.0.14.129
e.root-servers.net      internet address = 192.203.230.10
a.root-servers.net      internet address = 198.41.0.4
i.root-servers.net      internet address = 192.36.148.17
g.root-servers.net      internet address = 192.112.36.4
f.root-servers.net      internet address = 192.5.5.241
d.root-servers.net      internet address = 199.7.91.13

```

Figura 6 - Servidores de nome para o domínio "."

Os servidores de nome definidos para “.” são todos os nameserver apresentados na Figura 6. Nesta alínea foi possível verificar que à medida que o domínio se torna mais geral o número de nameservers existentes aumenta, tal como expectável.

**d. Existe o domínio eureka.software.? Será que eureka.software. é um host?**

```
> set q=NS
> eureka.software.
Server:      193.137.16.65
Address:     193.137.16.65#53

Non-authoritative answer:
eureka.software nameserver = ns-957.awsdns-55.net.
eureka.software nameserver = ns-312.awsdns-39.com.
eureka.software nameserver = ns-1241.awsdns-27.org.
eureka.software nameserver = ns-1624.awsdns-11.co.uk.

Authoritative answers can be found from:
ns-312.awsdns-39.com    internet address = 205.251.193.56
```

*Figura 7 – Query para eureka.software.*

Como é possível observar na Figura 7, ao questionar o domínio eureka.software acerca dos seus servidores de nome, verificamos que estão associados 4. Assim, podemos concluir que este domínio existe.

Com o intuito de verificar se eureka.software corresponde a um host elaboramos uma query para obter o endereço IP, tal como apresentado na figura seguinte.

```
> set q=A
> eureka.software.
Server:      193.137.16.65
Address:     193.137.16.65#53

Non-authoritative answer:
Name:   eureka.software
Address: 34.214.90.141
```

*Figura 8 – Query para eureka.software.*

Assim, com a existência de um endereço IP, podemos concluir que é um host.

**e. Qual é o servidor DNS primário definido para o domínio ami.pt.? Este servidor primário (master) aceita queries recursivas? Porquê?**

```

core@XubunCORE:~$ nslookup
> set q=SOA
> ami.pt
Server:          192.168.43.1
Address:         192.168.43.1#53

Non-authoritative answer:
ami.pt
      origin = ns1.dot2web.com
      mail addr = dc.dot2web.pt
      serial = 2019041005
      refresh = 3600
      retry = 7200
      expire = 1209600
      minimum = 86400

Authoritative answers can be found from:
ami.pt  nameserver = ns2.dot2web.com.
ami.pt  nameserver = ns1.dot2web.com.
ns1.dot2web.com internet address = 80.172.230.28
ns2.dot2web.com internet address = 54.36.137.213

```

Figura 9 – Query para ami.pt

De forma a obter o DNS primário definido para o domínio ami.pt, elaboramos uma query com o comando SOA. Como podemos verificar pela figura anterior, na parcela origin da secção *Non-authoritative answer*, o servidor DNS primário definido é o ns1.dot2web.com

```

> server ns1.dot2web.com
Default server: ns1.dot2web.com
Address: 80.172.230.28#53
> set q=NS
> ami.pt
Server:          ns1.dot2web.com
Address:         80.172.230.28#53

ami.pt  nameserver = ns2.dot2web.com.
ami.pt  nameserver = ns1.dot2web.com.
> uminho.pt
Server:          ns1.dot2web.com
Address:         80.172.230.28#53

** server can't find uminho.pt: REFUSED

```

Figura 10 - Queries com alteração do servidor

De modo a verificarmos se o servidor primário aceita queries recursivas alterou-se o servidor para o primário como podemos constatar em cima. Na figura 10 é possível verificar que para o domínio ami.pt obtemos resposta, dado que a informação se encontra nativamente. De seguida, averiguamos se obtemos resposta para uminho.pt e tendo em conta que tal não foi possível (REFUSED), podemos concluir que este não é recursivo.

- f. **Obtenha uma resposta “autoritativa” para a questão anterior.**

```

> server ns1.dot2web.com
Default server: ns1.dot2web.com
Address: 80.172.230.28#53
> ami.pt
Server:      ns1.dot2web.com
Address:     80.172.230.28#53

ami.pt
    origin = ns1.dot2web.com
    mail addr = dc.dot2web.pt
    serial = 2019040100
    refresh = 3600
    retry = 7200
    expire = 1209600
    minimum = 86400

```

Figura 11 - Exemplo de resposta autoritativa

De forma a obtermos uma resposta autoritativa, definimos como servidor default o ns1.dot2web.com, que corresponde ao servidor primário do domínio ami.pt, assim é possível realizar novamente a query.

**g. Onde são entregues as mensagens dirigidas a marcelo@presidencia.pt? E a guterres@onu.org?**

De modo a conhecer onde são entregues as mensagens dirigidas a um determinado email, foi necessário elaborar uma query DNS cujo tipo é MX.

```

core@XubunCORE:~$ nslookup
> set q=MX
> presidencia.pt
Server:      192.168.100.254
Address:     192.168.100.254#53

Non-authoritative answer:
presidencia.pt mail exchanger = 50 mail1.presidencia.pt.
presidencia.pt mail exchanger = 10 mail2.presidencia.pt.

Authoritative answers can be found from:
presidencia.pt nameserver = ns2.presidencia.pt.
presidencia.pt nameserver = ns02.fcn.pt.
presidencia.pt nameserver = ns1.presidencia.pt.
ns1.presidencia.pt internet address = 192.162.17.5
ns2.presidencia.pt internet address = 192.162.17.6
ns02.fcn.pt internet address = 193.136.2.228

```

Figura 12 - Query para o marcelo@presidencia.pt

As mensagens dirigidas a marcelo@presidencia.pt serão entregues a mail2.presidencia.pt, sendo este o servidor de email prioritário ou o alternativo mail1.presidencia.pt. Tal deve-se ao facto do primeiro apresentar uma prioridade de 10 e o segundo 50 e quanto menor este número maior será a prioridade. Isto pode ser observado na Figura 12.



```

> onu.org
Server:      192.168.100.254
Address:     192.168.100.254#53

Non-authoritative answer:
onu.org mail exchanger = 10 mail.onu.org.

Authoritative answers can be found from:
onu.org nameserver = cp.semillasl.com.
onu.org nameserver = ns01.semillasl.com.

```

Figura 13 - Query para o guterre@onu.org

Neste caso, como podemos verificar, serão entregues unicamente em mail.onu.org

- h. Que informação é possível obter acerca de [www.whitehouse.gov](http://www.whitehouse.gov)? Qual é o endereço IPv4 associado?

```

> set q=A
> www.whitehouse.gov
Server:      192.168.100.254
Address:     192.168.100.254#53

Non-authoritative answer:
www.whitehouse.gov canonical name = wildcard.whitehouse.gov.edgekey.net.
wildcard.whitehouse.gov.edgekey.net canonical name = e4036.dscb.akamaiedge.net.
Name: e4036.dscb.akamaiedge.net
Address: 23.10.65.110

```

Figura 14 – Query DNS sobre o domínio [www.whitehouse.gov](http://www.whitehouse.gov)

Através da Figura 14 foi possível verificar que este domínio é um alias do [wildcard.whitehouse.gov.edgekey.net](http://wildcard.whitehouse.gov.edgekey.net)., que por sua vez também é um alias do [e4036.dscb.akamaiedge.net](http://e4036.dscb.akamaiedge.net). Assim o endereço IPv4 associado é o 23.10.65.110 como podemos visualizar no campo *Address*.

- i. Consegue interrogar o DNS sobre o endereço IPv6 2001:690:a00:1036:1113::247 usando algum dos clientes DNS? Que informação consegue obter? Supondo que teve problemas com esse endereço, consegue obter um contacto do responsável por esse IPv6?

```

core@XubunCORE:~$ nslookup
> set q=PTR
> 2001:690:a00:1036:1113::247
Server:      193.137.16.65
Address:     193.137.16.65#53

Non-authoritative answer:
7.4.2.0.0.0.0.0.0.0.0.0.3.1.1.1.6.3.0.1.0.0.a.0.0.9.6.0.1.0.0.2.ip6.arpa n
ame = www.fccn.pt.

Authoritative answers can be found from:
0.9.6.0.1.0.0.2.ip6.arpa nameserver = ns03.fccn.pt.
0.9.6.0.1.0.0.2.ip6.arpa nameserver = ns01.fccn.pt.
0.9.6.0.1.0.0.2.ip6.arpa nameserver = ns02.fccn.pt.
ns02.fccn.pt internet address = 193.136.2.228
ns03.fccn.pt internet address = 138.246.255.249
ns01.fccn.pt internet address = 193.136.192.40
ns02.fccn.pt has AAAA address 2001:690:a80:4001::200
ns03.fccn.pt has AAAA address 2001:4ca0:106:0:250:56ff:fea9:3fd
ns01.fccn.pt has AAAA address 2001:690:a00:4001::200
>

```

Figura 15 – Query sobre o endereço IPv6 2001:690:a00:1036:1113::247



Na figura em cima verificamos que é possível interrogar o DNS sobre o endereço pretendido usando o cliente DNS nslookup, sendo possível obter a informação dos servidores de nome realizando uma query do tipo PTR.

```
> set q=SOA
> www.fccn.pt
Server:          193.137.16.65
Address:         193.137.16.65#53

Non-authoritative answer:
*** Can't find www.fccn.pt: No answer

Authoritative answers can be found from:
fccn.pt
    origin = ns01.fccn.pt
    mail addr = hostmaster.fccn.pt
    serial = 2019041101
    refresh = 21600
    retry = 7200
    expire = 1209600
    minimum = 14400
```

Figura 16 - Query para o www.fccn.pt

Com o intuito de obter um contacto responsável por esse IPv6, elaboramos uma query do tipo SOA para o domínio www.fccn.pt. Assim, no campo *origin* da Figura 16, podemos verificar que o contacto responsável por esse IPv6 é o ns01.fccn.pt.

- j. Os secundários usam o mecanismo designado por “Transferência de zona” para se atualizarem automaticamente a parte do primário, usando os parâmetros definidos no Record do tipo SOA do domínio. Descreve sucintamente esse mecanismo com base num exemplo concreto (ex: di.uminho.pt ou domínio cc.pt que vai ser criado na topologia virtual).

```
> set q=SOA
> di.uminho.pt
Server:          193.137.16.65
Address:         193.137.16.65#53

di.uminho.pt
    origin = dns.di.uminho.pt
    mail addr = dnsadmin.di.uminho.pt
    serial = 2019040803
    refresh = 28800
    retry = 7200
    expire = 28800
    minimum = 43200
>
```

Figura 17 - Query para o di.uminho.pt

O mecanismo de transferência de zona permite replicar a base de dados do servidor primário para o secundário. Para tal o secundário deverá requerer essa transferência ao primário. Tendo em conta os campos do Record do SOA do domínio di.uminho.pt, o

servidor secundário poderá aceder aos parâmetros temporais que irão permitir a sua atualização, bem como ao número de série da base de dados. Se este for o mesmo constatamos que não houve alterações à zona, caso contrário é necessário transferir novamente a zona através da utilização da query do tipo AXFR, obtendo a informação dos masters. No caso de o procedimento falhar o slave volta a tentar mais tarde, neste caso o servidor espera 7200 segundos para realizar uma nova tentativa, tal como se pode observar no parâmetro *retry*.

## PARTE II

Uma vez que a topologia fornecida possui quatro redes LAN tornou-se imperativo a implementação de uma zona para cada uma destas. Desta forma, o ficheiro `primario/named.conf` inclui cinco zonas. Como se trata do servidor primário todas estas são do tipo master. Por fim, foi necessário acrescentar uma cláusula que possibilitasse a transferência de dados para o servidor secundário através de `allow-transfer{10.2.2.3;}`.

```
include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default-zones";

zone "cc.pt" {
    type master;
    file "/home/core/primario/db.cc.pt";
    allow-transfer{ 10.2.2.3; };
};

zone "1.1.10.in-addr.arpa" {
    type master;
    file "/home/core/primario/db.1-1-10.rev";
    allow-transfer{ 10.2.2.3; };
};

zone "2.2.10.in-addr.arpa" {
    type master;
    file "/home/core/primario/db.2-2-10.rev";
    allow-transfer{ 10.2.2.3; };
};

zone "3.3.10.in-addr.arpa" {
    type master;
    file "/home/core/primario/db.3-3-10.rev";
    allow-transfer{ 10.2.2.3; };
};

zone "4.4.10.in-addr.arpa" {
    type master;
    file "/home/core/primario/db.4-4-10.rev";
    allow-transfer{ 10.2.2.3; };
};
```

Figura 18 - Ficheiro `primario/named.conf`

De seguida passamos à criação e configuração do ficheiro `primario/db.cc.pt`. Neste estabelecemos o servidor principal, o servidor `dns.cc.pt` e o grupo `51.cc.pt` como administrador. Como nameservers temos o `dns.cc.pt` e o `dns2.cc.pt` e os servidores de email, `Servidor2.cc.pt` que terá prioridade 20 e o `Servidor3.cc.pt` com prioridade 10.

Posteriormente para cada nó da topologia de rede associamos ao seu endereço IP o nome correspondente.

```
$TTL      604800
@         IN      SOA     Servidor1.cc.pt.  grupo51.cc.pt. (
                        2      ; Serial
                        604800  ; Refresh
                        86400   ; Retry
                        2419200 ; Expire
                        604800 ) ; Negative Cache TTL
;
@         IN      NS      Servidor1.cc.pt.
@         IN      NS      Urano.cc.pt.
;
dns       IN      CNAME   Servidor1.cc.pt.
dns2      IN      CNAME   Urano.cc.pt.
www       IN      CNAME   Servidor3.cc.pt.
mail      IN      CNAME   Servidor3.cc.pt.
pop       IN      CNAME   Servidor2.cc.pt.
imap      IN      CNAME   Servidor2.cc.pt.
Grupo51   IN      CNAME   Cliente1.cc.pt.
;
@         IN      MX      10      Servidor3
@         IN      MX      20      Servidor2
;
Cliente1  IN      A       10.4.4.1
Cliente2  IN      A       10.4.4.2
Cliente3  IN      A       10.4.4.3
Plutao    IN      A       10.2.2.1
Neptuno   IN      A       10.2.2.2
Urano     IN      A       10.2.2.3
Servidor1 IN      A       10.1.1.1
Servidor2 IN      A       10.1.1.2
Servidor3 IN      A       10.1.1.3
Alfa      IN      A       10.3.3.1
Beta      IN      A       10.3.3.2
Gama      IN      A       10.3.3.3
```

Figura 19 – Ficheiro primario/db.cc.pt

Depois criamos os ficheiros de domínios reverse que serão neste caso quatro tal como referido anteriormente.

```
$TTL      604800
@         IN      SOA     Servidor1.cc.pt.  grupo51.cc.pt. (
                        1      ; Serial
                        604800  ; Refresh
                        86400   ; Retry
                        2419200 ; Expire
                        604800 ) ; Negative Cache TTL
;
@         IN      NS      Servidor1.cc.pt.
@         IN      NS      Urano.cc.pt.
1         IN      PTR      Servidor1.cc.pt.
2         IN      PTR      Servidor2.cc.pt.
3         IN      PTR      Servidor3.cc.pt.
;
```

Figura 20 - primario/db.1-1-10.rev

## Testes

```

root@Beta: /tmp/pycore.40559/Beta.conf
root@Beta:/tmp/pycore.40559/Beta.conf# ping Servidor2
PING Servidor2.cc.pt (10.1.1.2) 56(84) bytes of data.
64 bytes from Servidor2.cc.pt (10.1.1.2): icmp_req=1 ttl=61 time=5.48 ms
64 bytes from Servidor2.cc.pt (10.1.1.2): icmp_req=2 ttl=61 time=5.20 ms
64 bytes from Servidor2.cc.pt (10.1.1.2): icmp_req=3 ttl=61 time=5.20 ms
64 bytes from Servidor2.cc.pt (10.1.1.2): icmp_req=4 ttl=61 time=5.21 ms
64 bytes from Servidor2.cc.pt (10.1.1.2): icmp_req=5 ttl=61 time=5.20 ms
64 bytes from Servidor2.cc.pt (10.1.1.2): icmp_req=6 ttl=61 time=5.22 ms
64 bytes from Servidor2.cc.pt (10.1.1.2): icmp_req=6 ttl=61 time=5.23 ms (DUP!)
64 bytes from Servidor2.cc.pt (10.1.1.2): icmp_req=7 ttl=61 time=5.21 ms
64 bytes from Servidor2.cc.pt (10.1.1.2): icmp_req=8 ttl=61 time=5.22 ms
64 bytes from Servidor2.cc.pt (10.1.1.2): icmp_req=9 ttl=61 time=5.22 ms
^C
--- Servidor2.cc.pt ping statistics ---
9 packets transmitted, 9 received, +1 duplicates, 0% packet loss, time 8015ms
rtt min/avg/max/mdev = 5.203/5.242/5.486/0.126 ms
root@Beta:/tmp/pycore.40559/Beta.conf#

```

Figura 21 - Ping do Beta para o Servidor2

```

root@Cliente3: /tmp/pycore.40559/Cliente3.conf
root@Cliente3:/tmp/pycore.40559/Cliente3.conf# ping Gama
PING Gama.cc.pt (10.3.3.3) 56(84) bytes of data.
64 bytes from Gama.cc.pt (10.3.3.3): icmp_req=1 ttl=62 time=10.2 ms
64 bytes from Gama.cc.pt (10.3.3.3): icmp_req=2 ttl=62 time=5.21 ms
64 bytes from Gama.cc.pt (10.3.3.3): icmp_req=2 ttl=62 time=5.22 ms (DUP!)
64 bytes from Gama.cc.pt (10.3.3.3): icmp_req=3 ttl=62 time=5.25 ms
64 bytes from Gama.cc.pt (10.3.3.3): icmp_req=4 ttl=62 time=5.20 ms
64 bytes from Gama.cc.pt (10.3.3.3): icmp_req=5 ttl=62 time=5.22 ms
64 bytes from Gama.cc.pt (10.3.3.3): icmp_req=6 ttl=62 time=5.19 ms
64 bytes from Gama.cc.pt (10.3.3.3): icmp_req=7 ttl=62 time=5.19 ms
^C
--- Gama.cc.pt ping statistics ---
7 packets transmitted, 7 received, +1 duplicates, 0% packet loss, time 6011ms
rtt min/avg/max/mdev = 5.193/5.845/10.258/1.670 ms
root@Cliente3:/tmp/pycore.40559/Cliente3.conf#

```

Figura 22 - Ping do Cliente3 para o Gama

```

root@Gama: /tmp/pycore.39820/Gama.conf
root@Gama:/tmp/pycore.39820/Gama.conf# nslookup - 10.1.1.1
> www.cc.pt
Server:      10.1.1.1
Address:     10.1.1.1#53

www.cc.pt    canonical name = Servidor3.cc.pt.
Name:  Servidor3.cc.pt
Address: 10.1.1.3
> set q=NS
> cc.pt
Server:      10.1.1.1
Address:     10.1.1.1#53

cc.pt    nameserver = Urano.cc.pt.
cc.pt    nameserver = Servidor1.cc.pt.
> set q=MX
> cc.pt
Server:      10.1.1.1
Address:     10.1.1.1#53

cc.pt    mail exchanger = 20 Servidor2.cc.pt.
cc.pt    mail exchanger = 10 Servidor3.cc.pt.
> set q=SOA
> cc.pt
Server:      10.1.1.1
Address:     10.1.1.1#53

cc.pt
      origin = Servidor1.cc.pt
      mail addr = grupo51.cc.pt
      serial = 2
      refresh = 604800
      retry = 86400
      expire = 2419200
      minimum = 604800
>

```

Figura 23 - Queries

```

root@Cliente1: /tmp/pycore.40559/Cliente1.conf
root@Cliente1:/tmp/pycore.40559/Cliente1.conf# nslookup www.cc.pt
Server:      10.1.1.1
Address:     10.1.1.1#53

www.cc.pt    canonical name = Servidor3.cc.pt.
Name:  Servidor3.cc.pt
Address: 10.1.1.3

root@Cliente1:/tmp/pycore.40559/Cliente1.conf#

```

Figura 24 - Query

## Conclusão

Com a realização do trabalho prático foi possível aprofundar os conhecimentos na área de serviço de resolução de nomes (DNS). Para tal, foi necessário pôr em prática os conceitos aprendidos nas aulas teóricas. Desta forma, foi possível responder a várias questões sobre o tema, bem como implementar e configurar servidores primário e secundário.

Neste trabalho a nossa maior dificuldade foi na segunda parte, que corresponde à implementação e configuração dos servidores, dado que foi o primeiro contacto estabelecido com esta área. Contudo, concluímos que realizamos o trabalho proposto com sucesso, uma vez que todas as funcionalidades foram implementadas.