



Universidade do Minho

Mestrado Integrado em Engenharia Informática

Comunicações por Computador

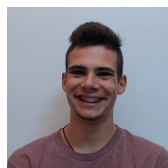
Trabalho Prático N.º.3 – Serviço de Resolução de
Nomes (DNS)

PL5 - Grupo 8

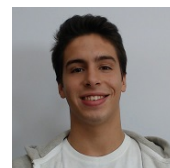
Ana Rita Rosendo
A84475



Gonçalo Esteves
A85731



Rui Oliveira
A83610



14 de Abril de 2020

Conteúdo

1	Questões e Respostas	3
1.1	Parte 1	3
1.1.1	Questão a	3
1.1.2	Questão b	3
1.1.3	Questão c	4
1.1.4	Questão d	5
1.1.5	Questão e	6
1.1.6	Questão f	6
1.1.7	Questão g	7
1.1.8	Questão h	8
1.1.9	Questão i	9
1.1.10	Questão j	10
2	Parte II	11
2.1	Criação e Configuração dos Ficheiros Necessários	11
2.2	Testes	13
3	Conclusão	15

1 Questões e Respostas

1.1 Parte 1

1.1.1 Questão a

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

O conteúdo do ficheiro `/etc/resolv.conf` está representado a seguir. Este contém diretivas que especificam um domínio de procura (neste caso, apenas o *Home*), usados para completar um nome de *query* quando não é fornecido um domínio de procura. Além disso, possui um *nameserver*, o endereço IP do servidor DNS. Neste caso, o endereço IP da máquina local.

```
core@XubunCORE:~$ cat /etc/resolv.conf
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 192.168.0.1
nameserver 2001:8a0:ff4b:d200::1
```

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

1.1.2 Questão b

Os servidores `www.sapo.pt.` e `www.yahoo.com.` têm endereços IPv6? Se sim, quais?

Os servidores `www.sapo.pt.` e `www.yahoo.com.` têm endereço IPv6. O endereço IPv6 do servidor `www.sapo.pt.` é `2001:8a0:2102:c:213:13:146:142`, os endereços do servidor `www.yahoo.com.` são `2a00:1288:110:1c::4` e `2a00:1288:110:1c::3`.

```
core@XubunCORE:~$ host www.sapo.pt.
www.sapo.pt has address 213.13.146.142
www.sapo.pt has IPv6 address 2001:8a0:2102:c:213:13:146:142
core@XubunCORE:~$ host www.yahoo.com.
www.yahoo.com is an alias for atsv2-fp-shed.wg1.b.yahoo.com.
atsv2-fp-shed.wg1.b.yahoo.com has address 87.248.98.7
atsv2-fp-shed.wg1.b.yahoo.com has address 87.248.98.8
atsv2-fp-shed.wg1.b.yahoo.com has IPv6 address 2a00:1288:110:1c::4
atsv2-fp-shed.wg1.b.yahoo.com has IPv6 address 2a00:1288:110:1c::3
```

Figura 2: Endereços IPv6

1.1.3 Questão c

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

O domínio “uminho.pt” tem os seguintes servidores de nomes definidos: “dns2.uminho.pt”, “ns02.fccn.pt”, “dns3.uminho.pt”, “dns.uminho.pt”.

```
core@XubunCORE:~$ nslookup
> set type=NS
> uminho.pt
Server:          192.168.1.254
Address:         192.168.1.254#53

Non-authoritative answer:
uminho.pt       nameserver = dns2.uminho.pt.
uminho.pt       nameserver = ns02.fccn.pt.
uminho.pt       nameserver = dns3.uminho.pt.
uminho.pt       nameserver = dns.uminho.pt.
```

Figura 3: Servidores de nomes para “uminho.pt”

O domínio “pt.” tem os seguintes servidores de nomes definidos: “f.dns.pt”, “g.dns.pt”, “ns.dns.br”, “ns2.nic.fr”, “h.dns.pt”, “a.dns.pt”, “b.dns.pt”, “c.dns.pt”, “d.dns.pt”, “e.dns.pt”.

```
core@XubunCORE:~$ nslookup
> set type=NS
> pt.
Server:          192.168.1.254
Address:         192.168.1.254#53

Non-authoritative answer:
pt      nameserver = f.dns.pt.
pt      nameserver = g.dns.pt.
pt      nameserver = ns.dns.br.
pt      nameserver = ns2.nic.fr.
pt      nameserver = h.dns.pt.
pt      nameserver = a.dns.pt.
pt      nameserver = b.dns.pt.
pt      nameserver = c.dns.pt.
pt      nameserver = d.dns.pt.
pt      nameserver = e.dns.pt.
```

Figura 4: Servidores de nomes para “pt.”

O domínio "." tem os seguintes servidores de nomes definidos: "k.root-servers.net", "f.root-servers.net", "g.root-servers.net", "j.root-servers.net", "e.root-servers.net", "h.root-servers.net", "a.root-servers.net", "d.root-servers.net", "m.root-servers.net", "c.root-servers.net", "i.root-servers.net", "b.root-servers.net", "l.root-servers.net".

```
core@XubunCORE:~$ nslookup
> set type=NS
> .
Server:          192.168.1.254
Address:         192.168.1.254#53

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

Figura 5: Servidores de nomes para "."

1.1.4 Questão d

Existe o domínio nice.software.? Será que nice.software. é um host ou um domínio ?

Através da query com tipo NS verificamos que "nice.software." tem servidores de nome, como tal, podemos afirmar que "nice.software." é um domínio.

Tal como pode ser observado na figura, "nice.software." também é um host uma vez que tem um endereço IP associado.

```
core@XubunCORE:~$ host nice.software.
nice.software has address 213.212.81.71
core@XubunCORE:~$ nslookup
> set type=NS
> nice.software.
Server:          192.168.1.254
Address:         192.168.1.254#53

Non-authoritative answer:
nice.software  nameserver = nsgbr.comlaude.co.uk.
nice.software  nameserver = nsusa.comlaude.net.
nice.software  nameserver = nssui.comlaude.ch.

Authoritative answers can be found from:
```

Figura 6: Query DNS para o domínio "nice.software."

1.1.5 Questão e

Qual é o servidor DNS primário definido para o domínio `msf.org`? Este servidor primário (`master`) aceita queries recursivas? Porquê?

O servidor primário definido para o domínio "`msf.org`." é "`ns1.dds.nl`". Outra forma de verificar a questão seria a partir do comando `nslookup` definindo o *type* como sendo `SOA`.

```
core@XubunCORE:~$ host -t soa msf.org.  
msf.org has SOA record ns1.dds.nl. postmaster.msf.org.  
1407464621 16384 2048 1048576 2560
```

Figura 7: Query DNS para o domínio "`msf.org`."

Em seguida, podemos observar o resultado de efetuar o comando `dig`, utilizando como parâmetro o servidor primário anteriormente obtido. Deste modo, podemos concluir que este servidor contém a *flag* "`ra`" (*Recursion Available*), o que significa que ele aceita queries recursivas.

```
core@XubunCORE:~$ dig ns1.dds.nl.  
  
; <<>> DiG 9.8.1-P1 <<>> ns1.dds.nl.  
;; global options: +cmd  
;; Got answer:  
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 31369  
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0  
  
;; QUESTION SECTION:  
;ns1.dds.nl.                IN      A  
  
;; ANSWER SECTION:  
ns1.dds.nl.                53438   IN      A      91.142.253.70
```

Figura 8: Query DNS para o servidor `ns1.dds.nl`.

1.1.6 Questão f

Obtenha uma resposta "autoritativa" para a questão anterior.

```
core@XubunCORE:~$ dig +short SOA msf.org. | cut -d ' ' -f1  
ns1.dds.nl.
```

Figura 9: Resposta autoritativa para a query dirigida a "`msf.org`."

1.1.7 Questão g

Onde são entregues as mensagens de correio eletrônico dirigidas aos presidentes **marcelo@presidencia.pt** e **bolsonaro@casacivil.gov.br**?

Utilizando a ferramenta *nslookup* e definindo o tipo da *query* como MX, por forma a identificar o servidor de *email* responsável por encaminhar os *emails* com destino a um dado domínio, poderemos facilmente determinar onde são entregues as mensagens.

No caso das mensagens dirigidas para "marcelo@presidencia.pt", estas serão entregues no servidor de *email* mais prioritário, que neste caso é "mail2.presidencia.pt", uma vez que este apresenta prioridade 10, por oposição ao segundo, que apresenta prioridade 50 - quanto menor o número, maior a prioridade.

Relativamente ao *email* "bolsonaro@casacivil.gov.br", as mensagens deste serão entregues no servidor "esa01.presidencia.gov.br".

```
core@XubunCORE:~$ nslookup
> set type=MX
> presidencia.pt
Server:          192.168.0.1
Address:         192.168.0.1#53

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

Authoritative answers can be found from:
> casacivil.gov.br.
Server:          192.168.0.1
Address:         192.168.0.1#53

Non-authoritative answer:
casacivil.gov.br      mail exchanger = 10 esa02.presidencia.gov.br.
casacivil.gov.br      mail exchanger = 5 esa01.presidencia.gov.br.

Authoritative answers can be found from:
```

Figura 10: *Query* DNS para os domínios pretendidos

1.1.8 Questão h

Que informação é possível obter, via DNS, acerca de `whitehouse.gov`?

De forma a obter informação relativa a "`www.whitehouse.gov`", podemos recorrer ao cliente DNS *host*, que nos indica que este domínio é um *alias* para "`wildcard.whitehouse.gov.edgekey.net`", que por sua vez é um *alias* para "`e4036.dscb.akamaiedge.net`". Podemos também afirmar que o endereço IP associado é 23.10.65.110.

```
core@XubunCORE:~$ host www.whitehouse.gov
www.whitehouse.gov is an alias for wildcard.whitehouse.gov.edgekey.net.
wildcard.whitehouse.gov.edgekey.net is an alias for e4036.dscb.akamaiedge.net.
e4036.dscb.akamaiedge.net has address 23.10.65.110
e4036.dscb.akamaiedge.net has IPv6 address 2001:15d8:2:28f::fc4
e4036.dscb.akamaiedge.net has IPv6 address 2001:15d8:2:28d::fc4
```

Figura 11: *Query* DNS para "`www.whitehouse.gov`" usando *host*

Poderíamos obter os mesmos resultados, recorrendo à ferramenta *nslookup*.

```
core@XubunCORE:~$ nslookup
> set type=A
> www.whitehouse.gov
Server:          192.168.0.1
Address:         192.168.0.1#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 12: *Query* DNS para "`www.whitehouse.gov`" usando *nslookup*

1.1.9 Questão 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?

É possível fazê-lo usando o cliente DNS nslookup. Usando a query com tipo PTR obtemos o domínio associado a um endereço IP onde podemos retirar informação acerca dos servidores de nome.

```
core@XubunCORE:~$ nslookup
> set type=PTR
> 2001:690:a00:1036:1113::247
Server:      192.168.1.254
Address:     192.168.1.254#53

Non-authoritative answer:
7.4.2.0.0.0.0.0.0.0.0.0.3.1.1.6.3.0.1.0.0.a.0.0.9.6.0.1.0.0.2.ip6.arpa      name = www.fccn.pt.

Authoritative answers can be found from:
6.3.0.1.0.0.a.0.0.9.6.0.1.0.0.2.ip6.arpa      nameserver = ns02.fccn.pt.
6.3.0.1.0.0.a.0.0.9.6.0.1.0.0.2.ip6.arpa      nameserver = ns01.fccn.pt.
```

Figura 13: *Query* para 2001:690:a00:1036:1113::247.

O domínio obtido foi o www.fccn.pt. De seguida, recorrendo à query com tipo SOA, obtemos os contactos responsáveis. Tal como pode ser observado na figura seguinte, o contacto responsável é ns01.fccn.pt.

```
> set type=SOA
> www.fccn.pt
Server:      192.168.1.254
Address:     192.168.1.254#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 = 2020040802
    refresh = 21600
    retry = 7200
    expire = 1209600
    minimum = 14400
```

Figura 14: *Query* para "www.fccn.pt".

1.1.10 Questão j

Os secundários usam um mecanismo designado por “Transferência de zona” para se atualizarem automaticamente a partir 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 o domínio cc.pt que vai ser criado na topologia virtual).

A transferência de zona é um mecanismo que permite replicar uma zona do banco de dados do servidor primário para um servidor secundário. Esta transferência é solicitada pelo servidor secundário ao servidor primário.

Com base nos parâmetros definidos no *Record* do tipo SOA do domínio “di.uminho.pt”, podemos retirar quatro diferentes aspetos relativamente a este mecanismo: primeiro, o parâmetro *refresh* representa o intervalo, em segundos, após o qual o servidor secundário irá pedir ao servidor primário uma atualização da sua informação; depois, o parâmetro *retry* representa a quantidade de segundos que o servidor secundário espera até tentar reconectar-se ao servidor primário após uma tentativa falhada; seguidamente, *expire* indica um limite temporal após o qual a informação de cache do primário, possuída pelo servidor secundário, se considera inválida. Por fim, o parâmetro *minimum* define o limite mínimo de tempo que deve passar até que o servidor secundário se atualize, desde a última atualização.

Deste modo, e através da observação da imagem em seguida demonstrada, podemos afirmar que a informação do servidor secundário nunca estará desatualizada por mais de 8h (28800 segundos); que este se tentará reconectar passado 2h (7200 segundos) após uma tentativa falhada; que a cache guardada pelo servidor secundário, proveniente do servidor primário, será válida por apenas 8h (28800 segundos); e que o servidor deverá aguardar sempre pelo menos 12h (43200 segundos) antes de se atualizar novamente.

```
core@XubunCORE:~$ nslookup
> set type=SOA
> di.uminho.pt
Server:          192.168.0.1
Address:         192.168.0.1#53

Non-authoritative answer:
di.uminho.pt
    origin = dns.di.uminho.pt
    mail addr = dnsadmin.di.uminho.pt
    serial = 2020021701
    refresh = 28800
    retry = 7200
    expire = 28800
    minimum = 43200
```

Figura 15: Resultado de uma *query* DNS de tipo SOA para “di.uminho.pt”

2 Parte II

2.1 Criação e Configuração dos Ficheiros Necessários

Após a realização dos primeiros passos enumerados no enunciado, chegamos à fase de alteração do ficheiro "primario/bind/named.conf". Uma vez que a topologia possui 4 redes LAN distintas, torna-se também necessário implementar 4 zonas diferentes, uma para cada rede. Deste modo, o ficheiro "primario/bind/named.conf" possui 5 zonas distintas, todas elas do tipo *master* uma vez que estamos a lidar com o servidor primário. Para além disto, é imperativo adicionar a cláusula *allow-transfer* 10.4.4.1, que permite a transferência de dados para o servidor secundário "Hermes".

```
zone "cc.pt" {
    type master;
    file "/home/core/primario/bind/db.cc.pt";
    allow-transfer{
        10.4.4.1;
    };
};

zone "1.1.10.in-addr.arpa" {
    type master;
    file "/home/core/primario/bind/db.1-1-10.rev";
    allow-transfer{
        10.4.4.1;
    };
};

zone "2.2.10.in-addr.arpa" {
    type master;
    file "/home/core/primario/bind/db.2-2-10.rev";
    allow-transfer{
        10.4.4.1;
    };
};

zone "3.3.10.in-addr.arpa" {
    type master;
    file "/home/core/primario/bind/db.3-3-10.rev";
    allow-transfer{
        10.4.4.1;
    };
};

zone "4.4.10.in-addr.arpa" {
    type master;
    file "/home/core/primario/bind/db.4-4-10.rev";
    allow-transfer{
        10.4.4.1;
    };
};
```

Figura 16: Ficheiro primario/bind/named.conf.

Posto isto, é necessário criar o ficheiro "db.cc.pt" e configurá-lo. Este será o ficheiro de dados do domínio de nomes.

Para começar, é necessário definir como servidor principal o servidor "dns.cc.pt" e como administrador "grupo08@cc.pt". Em seguida, definiram-se os *nameservers* (NS) "dns.cc.pt" e "dns2.cc.pt" bem como os servidores de e-mail (MX) "Serv3.cc.pt" e "Serv2.cc.pt", com prioridade 10 e 20, respetivamente. Posto isto, definimos o nome a que corresponde cada endereço IP da topologia de rede, bem com os diversos serviços existentes, recorrendo ao uso da clausula A, responsável por relacionar domínios com endereços IP. Graças a esta medida, torna-se possível efetuar comandos, como por exemplo um ping, para qualquer elemento da rede através do seu nome. Por forma a executar um ping para Serv1, podemos fazer "ping Serv1", ao invés de "ping 10.3.3.1".

```
$ORIGIN cc.pt.
$TTL      604800
@          IN      SOA      dns.cc.pt. grupo08.cc.pt. (
                        2          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )   ; Negative Cache TTL
;
@          IN      NS       Serv1.cc.pt.
@          IN      NS       Hermes.cc.pt.
@          IN      A        10.3.3.1
;
dns        IN      CNAME    Serv1.cc.pt.
dns2       IN      CNAME    Hermes.cc.pt.
;
Serv1      IN      A        10.3.3.1
Hermes     IN      A        10.4.4.1
Serv2      IN      A        10.3.3.2
Serv3      IN      A        10.3.3.3
Portatil1  IN      A        10.1.1.1
Atena      IN      A        10.4.4.3
Zeus       IN      A        10.4.4.2
Portatil2  IN      A        10.1.1.2
Portatil3  IN      A        10.1.1.3
Alfa       IN      A        10.2.2.1
Delta      IN      A        10.2.2.2
Omega      IN      A        10.2.2.3
;
@          IN      MX       10 Serv3
@          IN      MX       20 Serv2
;
pop        IN      CNAME    Serv2.cc.pt.
imap       IN      CNAME    Serv2.cc.pt.
;
www        IN      CNAME    Serv3.cc.pt.
mail       IN      CNAME    Serv3.cc.pt.
;
Grupo08    IN      CNAME    Portatil1.cc.pt.
```

Figura 17: Ficheiro primario/bind/db.cc.pt.

Por fim, é necessário criar e preencher todos os ficheiros de domínios *reverse*. Mais uma vez, torna-se necessário criar um ficheiro para cada rede LAN da topologia.

```
$TTL      604800
@         IN      SOA      dns.cc.pt. grupo08.cc.pt. (
                        1          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )   ; Negative Cache TTL
;
@         IN      NS       Serv1.cc.pt.
@         IN      NS       Hermes.cc.pt.
1         IN      PTR      Serv1.cc.pt.
2         IN      PTR      Serv2.cc.pt.
3         IN      PTR      Serv3.cc.pt.
```

Figura 18: Ficheiro primario/bind/db.3-3-10.rev.

2.2 Testes

Em seguida são apresentadas algumas imagens retiradas aquando da execução dos servidores criados, bem como de alguns comandos executados.

```
root@Serv1:/tmp/pycore.47441/Serv1.conf# sudo /usr/sbin/named -c /home/core/primario/bind/named.conf -g
14-Apr-2020 18:04:25.244 starting BIND 9.8.1-P1 -c /home/core/primario/bind/named.conf -g
14-Apr-2020 18:04:25.245 built with '--prefix=/usr' '--mandir=/usr/share/man' '--infodir=/usr/share/info'
 '--sysconfdir=/etc/bind' '--localstatedir=/var' '--enable-threads' '--enable-largefile' '--with-libtool'
 '--enable-shared' '--enable-static' '--with-openssl=/usr' '--with-gssapi=/usr' '--with-gnu-ld' '--with-g
eoip=/usr' '--enable-ipv6' 'CFLAGS=-fno-strict-aliasing -DDIG_SIGCHASE -O2' 'LDFLAGS=-Wl,-Bsymbolic-funct
ions -Wl,-z,relro' 'CPPFLAGS=-D_FORTIFY_SOURCE=2'
14-Apr-2020 18:04:25.245 adjusted limit on open files from 4096 to 1048576
14-Apr-2020 18:04:25.246 found 1 CPU, using 1 worker thread
14-Apr-2020 18:04:25.247 using up to 4096 sockets
14-Apr-2020 18:04:25.257 loading configuration from '/home/core/primario/bind/named.conf'
14-Apr-2020 18:04:25.258 reading built-in trusted keys from file '/etc/bind/bind.keys'
14-Apr-2020 18:04:25.260 using default UDP/IPv4 port range: [1024, 65535]
14-Apr-2020 18:04:25.261 using default UDP/IPv6 port range: [1024, 65535]
14-Apr-2020 18:04:25.265 listening on IPv6 interfaces, port 53
14-Apr-2020 18:04:25.266 listening on IPv4 interface lo, 127.0.0.1#53
14-Apr-2020 18:04:25.267 listening on IPv4 interface eth0, 10.3.3.1#53
14-Apr-2020 18:04:25.268 couldn't mkdir '/var/run/named': Permission denied
14-Apr-2020 18:04:25.269 generating session key for dynamic DNS
14-Apr-2020 18:04:25.270 couldn't mkdir '/var/run/named': Permission denied
14-Apr-2020 18:04:25.271 could not create /var/run/named/session.key
14-Apr-2020 18:04:25.273 failed to generate session key for dynamic DNS; permission denied
14-Apr-2020 18:04:25.275 sizing zone task pool based on 10 zones
14-Apr-2020 18:04:25.278 using built-in root key for view _default
```

Figura 19: Servidor Primário a correr.

```

root@Portatil1:/tmp/pycore.47441/Portatil1.conf# ping Atena
PING Atena.cc.pt (10.4.4.3) 56(84) bytes of data.
64 bytes from Atena.cc.pt (10.4.4.3): icmp_req=1 ttl=61 time=2.56 ms
64 bytes from Atena.cc.pt (10.4.4.3): icmp_req=2 ttl=61 time=0.469 ms
64 bytes from Atena.cc.pt (10.4.4.3): icmp_req=3 ttl=61 time=0.406 ms
64 bytes from Atena.cc.pt (10.4.4.3): icmp_req=4 ttl=61 time=0.537 ms
64 bytes from Atena.cc.pt (10.4.4.3): icmp_req=5 ttl=61 time=0.615 ms
64 bytes from Atena.cc.pt (10.4.4.3): icmp_req=6 ttl=61 time=0.676 ms
64 bytes from Atena.cc.pt (10.4.4.3): icmp_req=7 ttl=61 time=0.560 ms
64 bytes from Atena.cc.pt (10.4.4.3): icmp_req=8 ttl=61 time=0.604 ms

```

Figura 20: Ping para Atena.

```

root@Hermes:/tmp/pycore.47441/Hermes.conf# sudo /usr/sbin/named -c /home/core/secundario/bind/named.conf -g
14-Apr-2020 18:08:19.989 starting BIND 9.8.1-P1 -c /home/core/secundario/bind/named.conf -g
14-Apr-2020 18:08:19.991 built with '--prefix=/usr' '--mandir=/usr/share/man' '--infodir=/usr/share/info' '--sysconfdir=/etc/bind' '--localstatedir=/var' '--enable-threads' '--enable-largefile' '--with-libtool' '--enable-shared' '--enable-static' '--with-openssl=/usr' '--with-gssapi=/usr' '--with-gnu-ld' '--with-geoip=/usr' '--enable-ipv6' 'CFLAGS=-fno-strict-aliasing -DDIG_SIGCHASE -O2' 'LDFLAGS=-Wl,-Bsymbolic-functions -Wl,-z,relro' 'CPPFLAGS=-D_FORTIFY_SOURCE=2'
14-Apr-2020 18:08:19.992 adjusted limit on open files from 4096 to 1048576
14-Apr-2020 18:08:19.993 found 1 CPU, using 1 worker thread
14-Apr-2020 18:08:19.994 using up to 4096 sockets
14-Apr-2020 18:08:20.004 loading configuration from '/home/core/secundario/bind/named.conf'
14-Apr-2020 18:08:20.005 reading built-in trusted keys from file '/etc/bind/bind.keys'
14-Apr-2020 18:08:20.006 using default UDP/IPv4 port range: [1024, 65535]
14-Apr-2020 18:08:20.007 using default UDP/IPv6 port range: [1024, 65535]
14-Apr-2020 18:08:20.012 listening on IPv6 interfaces, port 53
14-Apr-2020 18:08:20.013 listening on IPv4 interface lo, 127.0.0.1#53
14-Apr-2020 18:08:20.014 listening on IPv4 interface eth0, 10.4.4.1#53
14-Apr-2020 18:08:20.015 couldn't mkdir '/var/run/named': Permission denied
14-Apr-2020 18:08:20.016 generating session key for dynamic DNS
14-Apr-2020 18:08:20.016 couldn't mkdir '/var/run/named': Permission denied
14-Apr-2020 18:08:20.017 could not create /var/run/named/session.key
14-Apr-2020 18:08:20.018 failed to generate session key for dynamic DNS: permission denied
14-Apr-2020 18:08:20.018 sizing zone task pool based on 10 zones
14-Apr-2020 18:08:20.022 using built-in root key for view _default

```

Figura 21: Servidor Secundário a correr.

```

root@Delta:/tmp/pycore.47441/Delta.conf# nslookup - 10.4.4.1
> www.cc.pt
Server:          10.4.4.1
Address:         10.4.4.1#53

www.cc.pt        canonical name = Serv3.cc.pt.
Name:   Serv3.cc.pt
Address: 10.3.3.3

```

Figura 22: Execução do comando nslookup.

3 Conclusão

Neste trabalho foi abordado o tema "Serviço de Resolução de Nomes (DNS)" o que nos permitiu expandir os nossos conhecimentos nessa área. Ao longo da sua resolução, foram-nos expostas diversas dificuldades, dificuldades essas que foram ultrapassadas devido ao trabalho de pesquisa realizado e ao conhecimento adquirido ao longo das aulas. Como resultado final, foram implementados os servidores propostos, funcionais.