

UNIVERSIDADE DO MINHO
LICENCIATURA EM ENGENHARIA INFORMÁTICA

COMUNICAÇÕES POR COMPUTADOR

Trabalho Prático 1 PROTOCOLOS DE CAMADA DE TRANSPORTE

Grupo 49



António Fonseca
a93167



Diogo Rebelo
a93278



Henrique Alvelos
a93316

23 de novembro de 2021

Conteúdo

1	Introdução	4
2	Questões e Respostas	5
2.1	Alínea a)	5
2.1.1	O ficheiro <code>/etc/resolv.conf</code>	5
2.2	Alínea b)	6
2.2.1	<i>IP - Internet Protocol</i>	6
2.2.2	IPv4 vs IPv6	6
2.3	Alínea c)	8
2.4	Alínea d)	9
2.5	Alínea e)	10
2.6	Alínea f)	11
2.7	Alínea g)	12
2.8	Alínea h)	13
2.9	Alínea i)	15
2.10	Alínea j)	16
2.10.1	Zona de Transferência	16
2.10.2	A operação de Transferência	16
2.10.3	Vulnerabilidade e Prevenção	17
3	Parte II	18
3.1	Modo de resolução	18
3.2	Preparativos especiais para o CORE	18
3.2.1	Passo 1) replicar ficheiros de configuração	18
3.2.2	Passo 2) ver se o servidor DNS pré-instalado está em execução, parando-o de seguida se necessário	18
3.2.3	Passo 3) reconfigurar apparmor para permitir que <code>/usr/sbin/named</code> aceda a ficheiros noutros locais	18
3.3	Configuração do servidor primário	20
3.3.1	Editar o ficheiro <code>/etc/hosts</code>	20
3.3.2	Editar o ficheiro <code>/CC-TP3/primario/named.conf.options</code>	20
3.3.3	Editar o ficheiro <code>/CC-TP3/primario/named.conf</code>	20
3.3.4	Criar o ficheiro de dados do domínio	21
3.3.5	Criar os ficheiros reversos	22
3.3.6	Testar configurações e ficheiros de dados	23
3.3.7	Executar o servidor	25
3.4	Configuração do cliente e teste do primário	26
3.5	Configuração do servidor secundário	28
3.5.1	Editar o ficheiro <code>named.conf.options</code>	28
3.5.2	Editar o ficheiro <code>named.conf</code>	28
3.5.3	Teste dos ficheiros de configuração	29
3.5.4	Executar servidor e abrir a bash do nó Golfinho	29
4	Conclusões	30

Lista de Figuras

1	Conteúdo do ficheiro <code>/etc/resolv.conf</code>	5
2	Informação dos endereços de IP	6
3	Listagem dos endereços IPv4 e IPv6	7
4	Name servers de “gov.pt.”	8
5	Name servers de “.”	8
6	Informação obtida com o comando <code>host</code>	9
7	Obtenção do servidor primário	10
8	Evidência da flag que determina recursividade	10
9	Obtenção de uma resposta autoritativa	11
10	Informação obtida através da query MX	12
11	Nome dos servidores associados a “gov.pt”	13
12	Informação autoritativa sobre “gov.pt”	13
13	Informação obtida com o comando <code>dig</code>	14
14	Informação obtida com a query PTR	15
15	Obtenção da informação com o comando <code>dig</code>	17
16	Obtenção da informação com o comando <code>host</code>	17
17	Prova de reinício do servidor	18
18	Ficheiro <code>usr.sbin.named</code> com as respetivas permissões adicionadas.	19
19	Ficheiro <code>/etc/hosts</code> alterado.	20
20	Ficheiro <code>/CC-TP3/primario/named.conf.options</code> alterado.	20
21	Ficheiro <code>/CC-TP3/primario/named.conf</code> alterado.	21
22	Ficheiro <code>/CC-TP3/primario/db.cc.pt</code> criado.	22
23	Ficheiro <code>db.1-1-10.rev</code> criado.	22
24	Ficheiro <code>db.2-2-10.rev</code> criado.	23
25	Ficheiro <code>db.3-3-10.rev</code> criado.	23
26	Teste de configuração e dos ficheiros	24
27	Teste do servidor	25
28	Questionar servidor sobre o domínio <code>www.cc.pt.</code>	26
29	Teste efetuado na bash requisitada no nó do Servidor1	27
30	Query efetuada no Portátil1.	27
31	Ficheiro <code>named.conf.options</code> alterado.	28
32	Ficheiro <code>named.conf</code> alterado.	28
33	Testes realizados aos ficheiros criados.	29
34	Testes realizados no nó Golfinho.	29
35	Testes lookup em qualquer nó da topologia.	29

1 Introdução

O presente relatório tenta explorar de uma forma mais profunda alguns conceitos primordiais sobre o Sistema de Resolução de Nomes (DNS), com a criação de um servidor primário e secundário. Naturalmente, tendo sempre como base princípios teóricos, socorremo-nos de uma abordagem mais prática e com o auxílio de pesquisas para o realizar com sucesso. Assim sendo, este relatório compreende um conjunto de objetivos fundamentais:

- Compreender o método de resolução de Nomes, num contexto DNS;
- Criar um servidor primário e secundário;
- Compreender a configuração base dos diferentes ficheiros utilizados na criação destes servidores;
- Lecionar de um modo mais profundo a utilização dos diferentes comandos em contexto de busca de informação sobre um servidor DNS (dig, nslookup, host, etc.);
- Aprofundar conceitos sobre a consulta de serviços de nome;
- Ganhar um maior traquejo em relação à utilização da *Virtual Box* e *Core*.

2 Questões e Respostas

Nesta secção constam todas as questões colocadas e respetivas respostas. Optamos por introduzir alguma informação que consideramos relevante para uma compreensão bem conseguida.

Comece-se, assim sendo, pela **Parte I**.

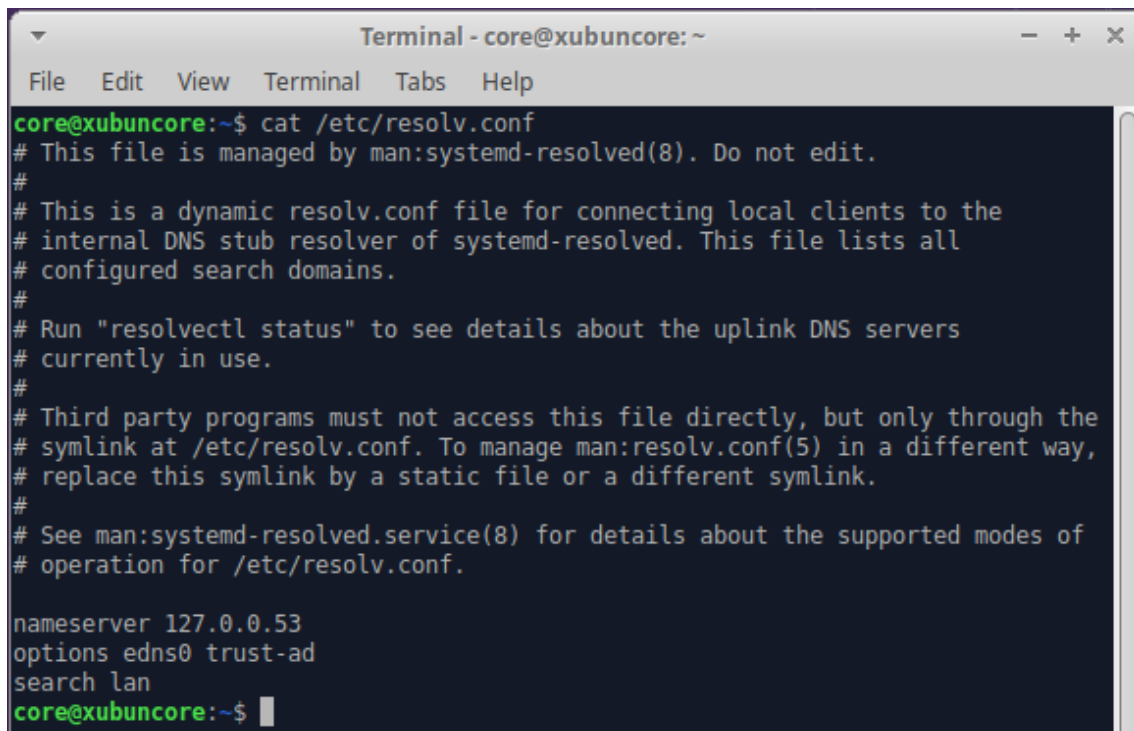
2.1 Alínea a)

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

2.1.1 O ficheiro `/etc/resolv.conf`

De um modo geral, este ficheiro `resolv.conf` é utilizado em vários sistemas operativos como forma de configurar o *DNS Resolver*, isto é, o nome dos servidores DNS. No sentido de compreender melhor a utilização deste ficheiro, segue abaixo uma lista mais completa com os pontos primordiais alusivos ao seu conteúdo e à sua configuração:

- É um ficheiro de **Plain-text**, ou seja, de texto, normalmente criado pelo administrador de rede ou pelas aplicações que gerem a configuração de tarefas no próprio sistema;
- Contém o conjunto de parâmetros utilizados pelo *DNS Resolver* para determinar o DNS utilizado por defeito, num contexto de informação incompleta;
- Contém a informação lida pelo *Resolver* na primeira vez que é invocado por um processo;
- É configurado de modo a ser *human-friendly*, ou seja, facilmente lido pelo ser humano;
- Contém as diretrizes de pesquisa de domínios utilizadas para transformar o nome de uma query dada num nome absoluto de domínio, quando nenhum sufixo deste é fornecido;
- Contém a lista de Endereços de IP dos servidores para resolução.

A screenshot of a terminal window titled "Terminal - core@xubuncore: ~". The window has a menu bar with "File", "Edit", "View", "Terminal", "Tabs", and "Help". The terminal shows the command `cat /etc/resolv.conf` being executed. The output is as follows:

```
core@xubuncore:~$ cat /etc/resolv.conf
# This file is managed by man:systemd-resolved(8). Do not edit.
#
# This is a dynamic resolv.conf file for connecting local clients to the
# internal DNS stub resolver of systemd-resolved. This file lists all
# configured search domains.
#
# Run "resolvectl status" to see details about the uplink DNS servers
# currently in use.
#
# Third party programs must not access this file directly, but only through the
# symlink at /etc/resolv.conf. To manage man:resolv.conf(5) in a different way,
# replace this symlink by a static file or a different symlink.
#
# See man:systemd-resolved.service(8) for details about the supported modes of
# operation for /etc/resolv.conf.

nameserver 127.0.0.53
options edns0 trust-ad
search lan
core@xubuncore:~$
```

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

2.2 Alínea b)

| Os servidores `www.di.uminho.pt.` e `www.europa.eu.` têm endereços IPv6? Se sim, quais?

2.2.1 IP - Internet Protocol

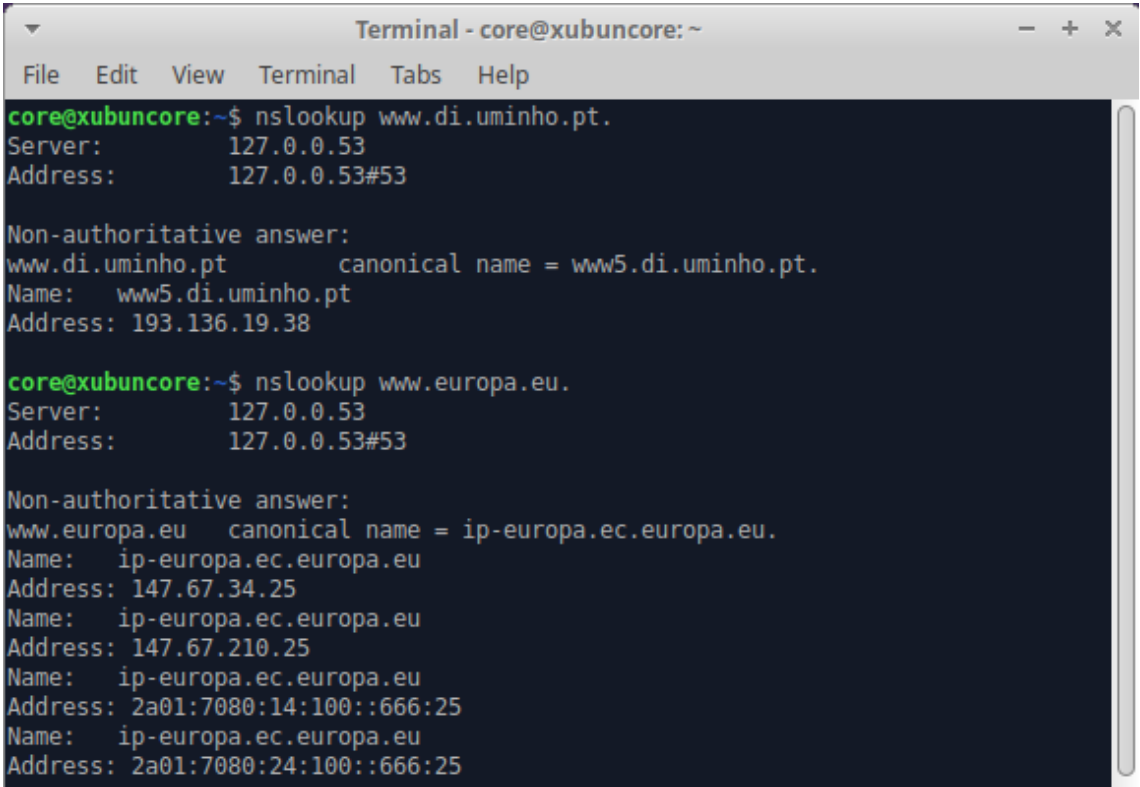
IP é um protocolo que ajuda os computadores a comunicarem entre si, através de uma rede. Todos os dispositivos que utilizam este protocolo têm o seu endereço de IP específico, o qual funciona como um rótulo: conjunto de regras importantes na envio e receção de pacotes. Cada um destes pacotes possui informações de IP. Assim, garantimos que estes chegam ao ponto certo. Já vimos que cada domínio na Internet tem o seu endereço de IP, que o identifica de forma única. A maioria dos dispositivos combina IP com TCP, o que permite, como vimos no trabalho anterior, a conexão entre uma fonte e um destino (um cliente e um servidor).

2.2.2 IPv4 vs IPv6

O protocolo descrito anteriormente é constituído por duas versões (como o tal “v” sugere):

- **IPv4:** é a versão mais conhecida para identificar dispositivos numa rede. Este utiliza endereços de 32 bits, permitindo fornecer 2^{32} endereços.
- **IPv6 ou IPng (next generation)** é a versão mais recente do protocolo e surge perante a necessidade de ter mais endereços de Internet. Este utiliza endereços de 128 bits, permitindo fornecer 2^{138} endereços. IPv6 usa tantos números como letras.

Utilizou-se o comando `nslookup` para perceber se os servidores em questão utilizam IPv4 e/ou IPv6. Aparecem listados vários endereços e só pelo seu formato é fácil verificar a sua versão:



```
Terminal - core@xubuncore: ~
File Edit View Terminal Tabs Help

core@xubuncore:~$ nslookup www.di.uminho.pt.
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
www.di.uminho.pt      canonical name = www5.di.uminho.pt.
Name:   www5.di.uminho.pt
Address: 193.136.19.38

core@xubuncore:~$ nslookup www.europa.eu.
Server:      127.0.0.53
Address:     127.0.0.53#53

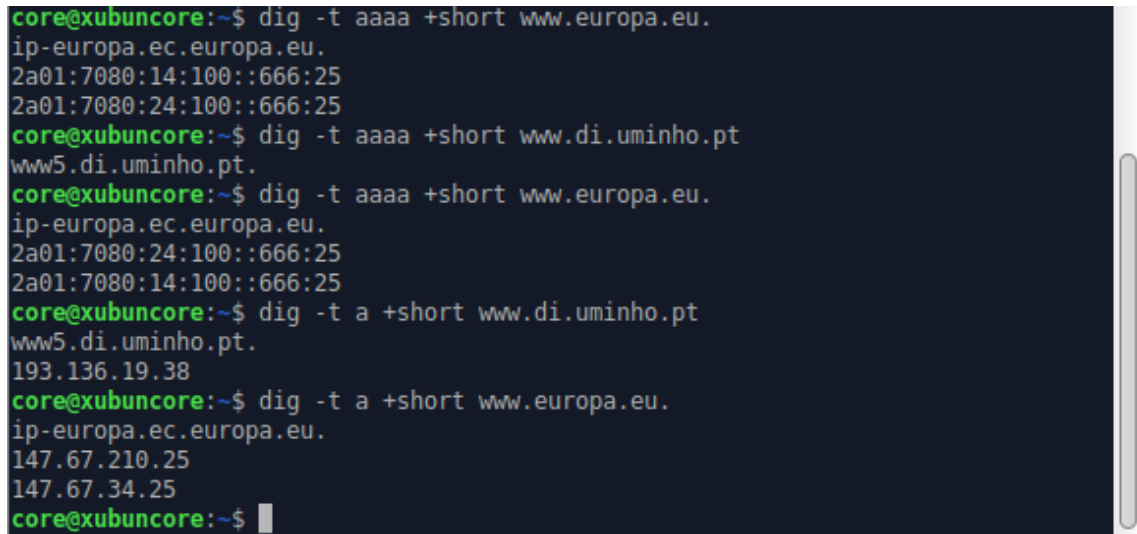
Non-authoritative answer:
www.europa.eu        canonical name = ip-europa.ec.europa.eu.
Name:   ip-europa.ec.europa.eu
Address: 147.67.34.25
Name:   ip-europa.ec.europa.eu
Address: 147.67.210.25
Name:   ip-europa.ec.europa.eu
Address: 2a01:7080:14:100::666:25
Name:   ip-europa.ec.europa.eu
Address: 2a01:7080:24:100::666:25
```

Figura 2: Informação dos endereços de IP

Compreende-se que o servidor `www.di.uminho.pt.` possui apenas IPv4 (um endereço), não possuindo IPv6. Já o servidor `www.europa.eu.` possui IPv4 (dois endereços) e IPv6 (dois endereços), também visível pelo formato dos endereços apresentados. Contudo, se não tivéssemos a certeza, uma forma fácil de verificar a existência destes protocolos é utilizar os seguintes comandos:

`dig -t aaaa +short <nome do servidor>` (fornece o(s) endereço(s) IPv6);
`dig -t a +short <nome do servidor>` (fornece o(s) endereço(s) IPv4).

O comando `host <nome do servidor>` também lista de imediato estes endereços.



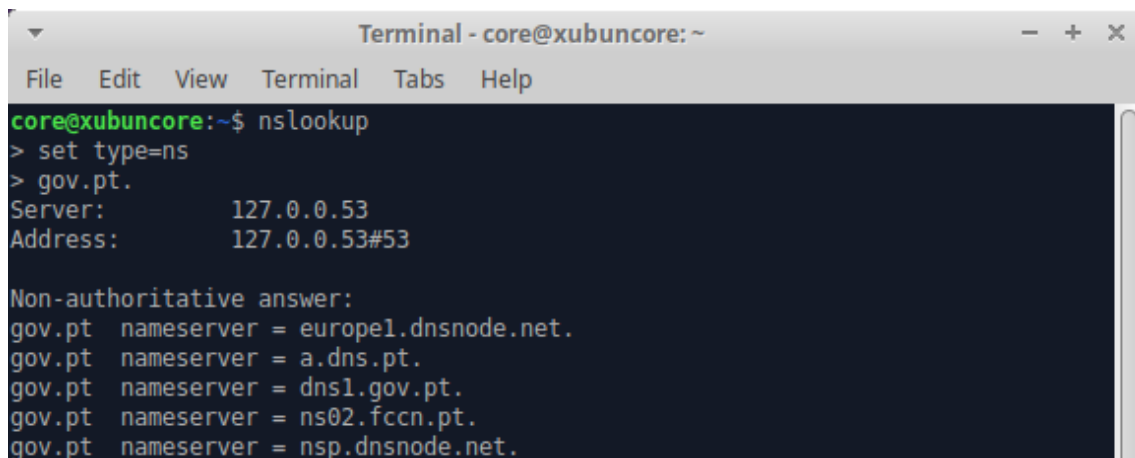
```
core@xubuncore:~$ dig -t aaaa +short www.europa.eu.  
ip-europa.ec.europa.eu.  
2a01:7080:14:100::666:25  
2a01:7080:24:100::666:25  
core@xubuncore:~$ dig -t aaaa +short www.di.uminho.pt  
www5.di.uminho.pt.  
core@xubuncore:~$ dig -t aaaa +short www.europa.eu.  
ip-europa.ec.europa.eu.  
2a01:7080:24:100::666:25  
2a01:7080:14:100::666:25  
core@xubuncore:~$ dig -t a +short www.di.uminho.pt  
www5.di.uminho.pt.  
193.136.19.38  
core@xubuncore:~$ dig -t a +short www.europa.eu.  
ip-europa.ec.europa.eu.  
147.67.210.25  
147.67.34.25  
core@xubuncore:~$
```

Figura 3: Listagem dos endereços IPv4 e IPv6

2.3 Alínea c)

| Quais os servidores de nomes definidos para os domínios: “gov.pt.” e “.”?

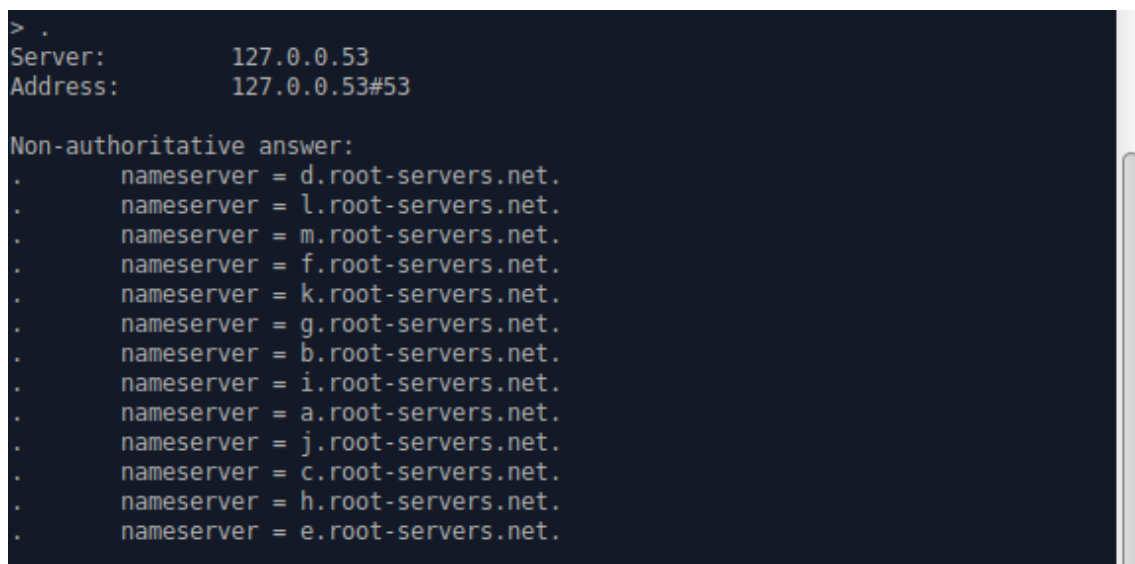
Para isso, basta escrever no terminal `nslookup`, definir `type set=ns` e fornecer o nome do domínio `gov.pt.` e `.`, como se mostra de seguida:

A terminal window titled "Terminal - core@xubuncore: ~" with a menu bar (File, Edit, View, Terminal, Tabs, Help). The prompt is "core@xubuncore:~\$". The user enters "nslookup", then "> set type=ns", and then "> gov.pt.". The output shows the server and address as 127.0.0.53. Below this, it says "Non-authoritative answer:" followed by a list of nameservers for gov.pt.: europol.dnsnode.net., a.dns.pt., dns1.gov.pt., ns02.fccn.pt., and nsp.dnsnode.net.

```
core@xubuncore:~$ nslookup
> set type=ns
> gov.pt.
Server:          127.0.0.53
Address:         127.0.0.53#53

Non-authoritative answer:
gov.pt  nameserver = europol.dnsnode.net.
gov.pt  nameserver = a.dns.pt.
gov.pt  nameserver = dns1.gov.pt.
gov.pt  nameserver = ns02.fccn.pt.
gov.pt  nameserver = nsp.dnsnode.net.
```

Figura 4: Name servers de “gov.pt.”

A terminal window showing the continuation of the nslookup command. The user enters "> ." after the previous output. The output shows the server and address as 127.0.0.53. Below this, it says "Non-authoritative answer:" followed by a list of nameservers for the root domain: d.root-servers.net., l.root-servers.net., m.root-servers.net., f.root-servers.net., k.root-servers.net., g.root-servers.net., b.root-servers.net., i.root-servers.net., a.root-servers.net., j.root-servers.net., c.root-servers.net., h.root-servers.net., and e.root-servers.net.

```
> .
Server:          127.0.0.53
Address:         127.0.0.53#53

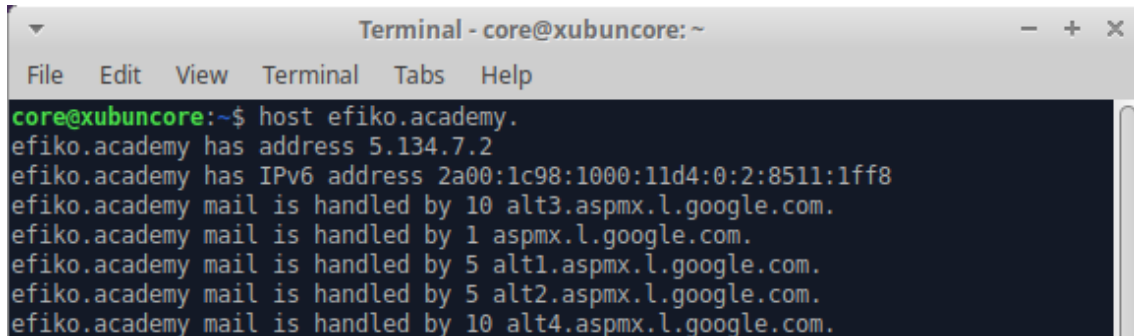
Non-authoritative answer:
.       nameserver = d.root-servers.net.
.       nameserver = l.root-servers.net.
.       nameserver = m.root-servers.net.
.       nameserver = f.root-servers.net.
.       nameserver = k.root-servers.net.
.       nameserver = g.root-servers.net.
.       nameserver = b.root-servers.net.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

Figura 5: Name servers de “.”

2.4 Alínea d)

Existe o domínio efiko.academy.? Com base na informação obtida do DNS, nomeadamente os registos associados a esse nome, diga se o considera um host ou um domínio de nomes.

Ao utilizar o comando `host`, descrito anteriormente, verifica-se que o domínio em questão possui um endereço de IP a si associado. Deste modo, conclui-se que é um host.

A screenshot of a terminal window titled "Terminal - core@xubuncore: ~". The window has a menu bar with "File", "Edit", "View", "Terminal", "Tabs", and "Help". The terminal shows the command `core@xubuncore:~$ host efiko.academy.` and its output: `efiko.academy has address 5.134.7.2`, `efiko.academy has IPv6 address 2a00:1c98:1000:11d4:0:2:8511:1ff8`, and five lines indicating mail handling by different `alt` subdomains of `aspmx.l.google.com.`.

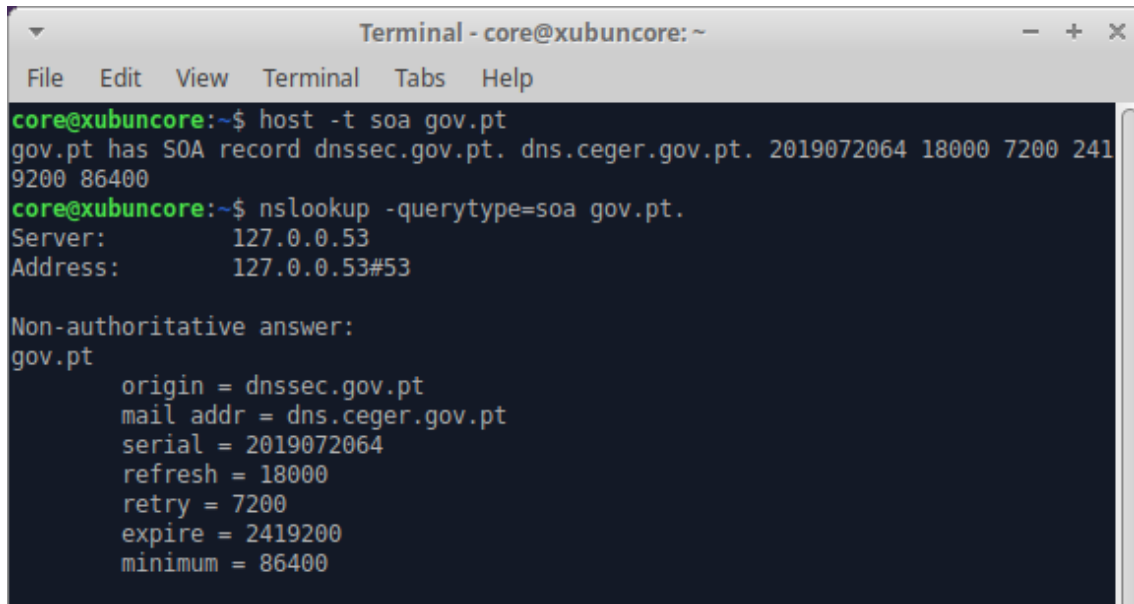
```
core@xubuncore:~$ host efiko.academy.
efiko.academy has address 5.134.7.2
efiko.academy has IPv6 address 2a00:1c98:1000:11d4:0:2:8511:1ff8
efiko.academy mail is handled by 10 alt3.aspmx.l.google.com.
efiko.academy mail is handled by 1 aspmx.l.google.com.
efiko.academy mail is handled by 5 alt1.aspmx.l.google.com.
efiko.academy mail is handled by 5 alt2.aspmx.l.google.com.
efiko.academy mail is handled by 10 alt4.aspmx.l.google.com.
```

Figura 6: Informação obtida com o comando `host`

2.5 Alínea e)

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

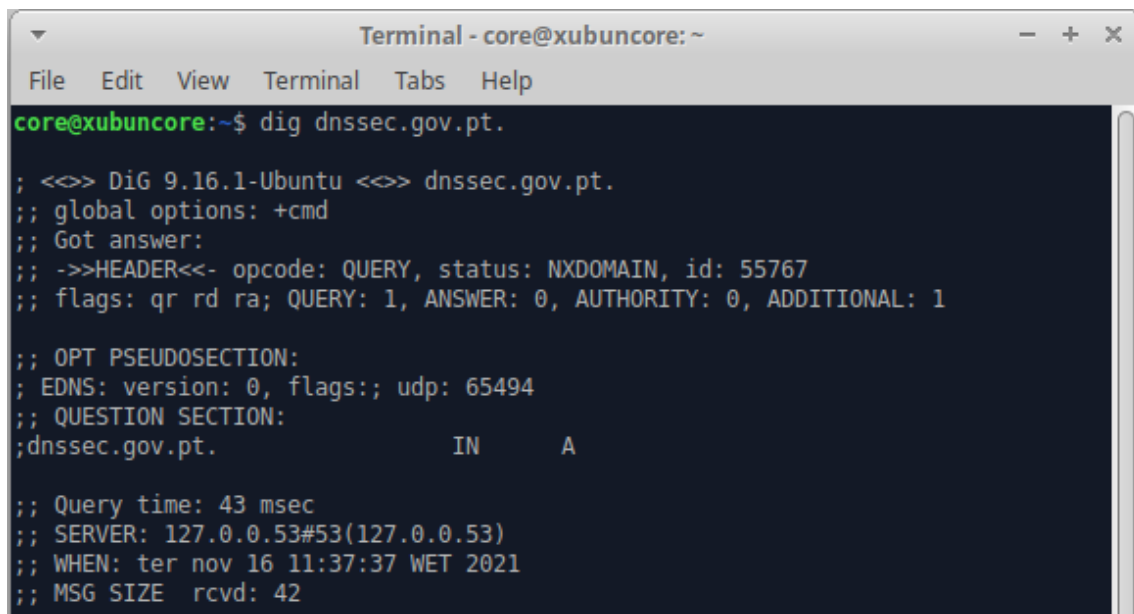
Com a ajuda do comando `host -t soa gov.pt.`, descobrimos que o servidor de DNS primário para gov.pt é `dnssec.gov.pt.` De seguida, com o `dig`, observa-se que uma das flags é “ra” pelo que, o master aceita queries recursivas. Uma outra forma de perceber, seria testar e verificar se a nossa query era recusada.



```
Terminal - core@xubuncore: ~
File Edit View Terminal Tabs Help
core@xubuncore:~$ host -t soa gov.pt
gov.pt has SOA record dnssec.gov.pt. dns.ceger.gov.pt. 2019072064 18000 7200 2419200 86400
core@xubuncore:~$ nslookup -querytype=soa gov.pt.
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
gov.pt
    origin = dnssec.gov.pt
    mail addr = dns.ceger.gov.pt
    serial = 2019072064
    refresh = 18000
    retry = 7200
    expire = 2419200
    minimum = 86400
```

Figura 7: Obtenção do servidor primário



```
Terminal - core@xubuncore: ~
File Edit View Terminal Tabs Help
core@xubuncore:~$ dig dnssec.gov.pt.

; <<>> DiG 9.16.1-Ubuntu <<>> dnssec.gov.pt.
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 55767
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:;, udp: 65494
;; QUESTION SECTION:
;dnssec.gov.pt.                IN      A

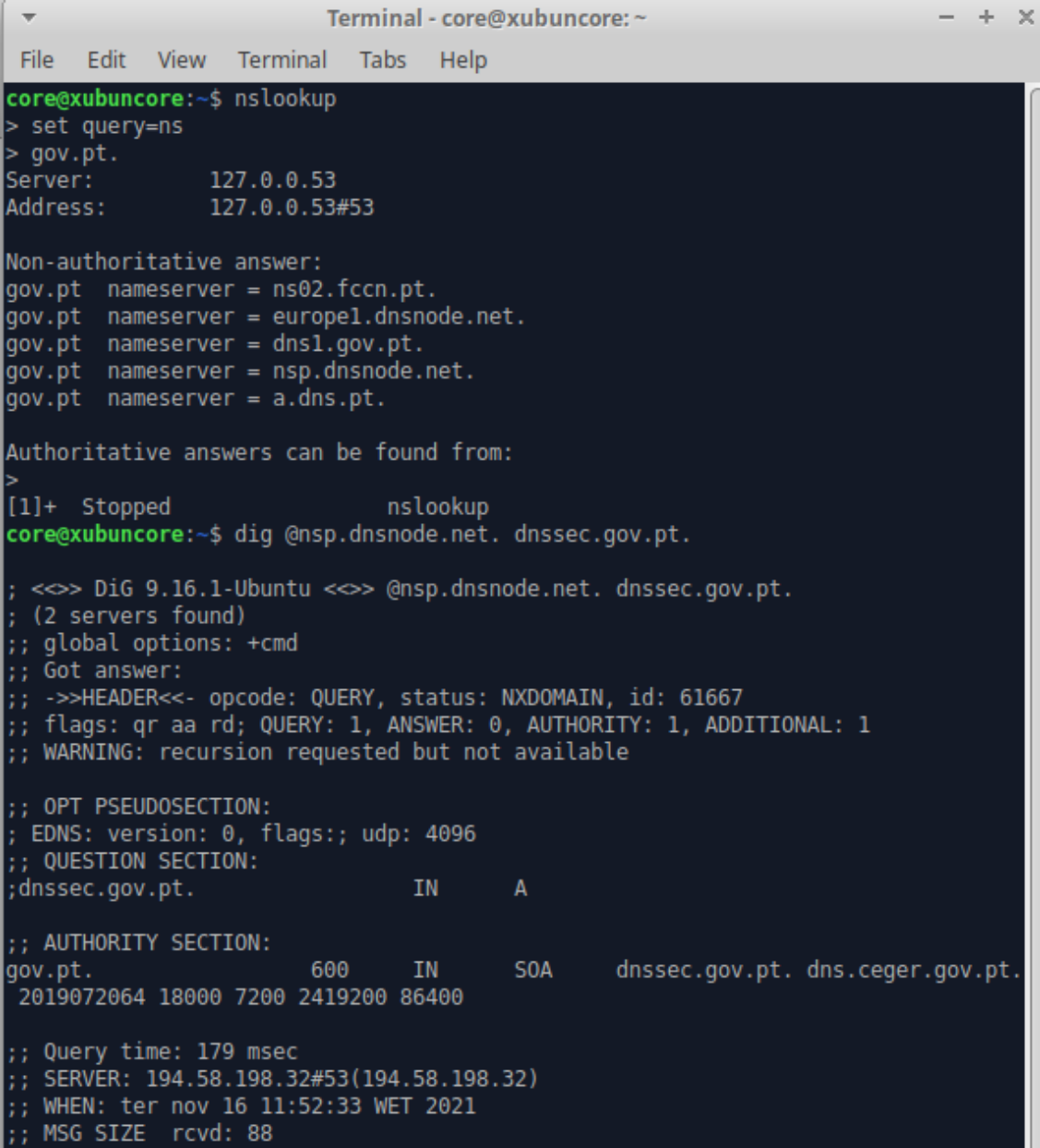
;; Query time: 43 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: ter nov 16 11:37:37 WET 2021
;; MSG SIZE rcvd: 42
```

Figura 8: Evidência da flag que determina recursividade

2.6 Alínea f)

| Obtenha uma resposta “autoritativa” para a questão anterior.

Para obter uma resposta autoritativa é necessário comunicar diretamente com um servidor autoritativo de "gov.pt", ou seja, com o servidor que controla o domínio em questão. Para isso, urge saber os name servers com autoridade sobre gov.pt., com o auxílio do comando `nslookup` (query do tipo NS). De seguida, utilizamos o comando `dig` com o parâmetro "@nsp.dnsnode.net"



```
Terminal - core@xubuncore: ~
File Edit View Terminal Tabs Help

core@xubuncore:~$ nslookup
> set query=ns
> gov.pt.
Server:          127.0.0.53
Address:         127.0.0.53#53

Non-authoritative answer:
gov.pt  nameserver = ns02.fccn.pt.
gov.pt  nameserver = europel.dnsnode.net.
gov.pt  nameserver = dns1.gov.pt.
gov.pt  nameserver = nsp.dnsnode.net.
gov.pt  nameserver = a.dns.pt.

Authoritative answers can be found from:
>
[1]+  Stopped                  nslookup
core@xubuncore:~$ dig @nsp.dnsnode.net. dnssec.gov.pt.

; <<> DiG 9.16.1-Ubuntu <<> @nsp.dnsnode.net. dnssec.gov.pt.
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 61667
;; flags: qr aa rd; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1
;; WARNING: recursion requested but not available

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;dnssec.gov.pt.                IN      A

;; AUTHORITY SECTION:
gov.pt.                        600     IN      SOA     dnssec.gov.pt. dns.ceger.gov.pt.
2019072064 18000 7200 2419200 86400

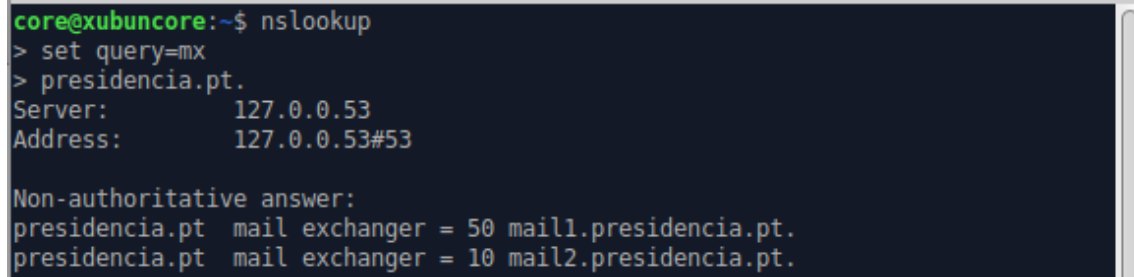
;; Query time: 179 msec
;; SERVER: 194.58.198.32#53(194.58.198.32)
;; WHEN: ter nov 16 11:52:33 WET 2021
;; MSG SIZE rcvd: 88
```

Figura 9: Obtenção de uma resposta autoritativa

2.7 Alínea g)

| Onde são entregues as mensagens de correio eletrónico dirigidas a marcelo@presidencia.pt?

Surge, agora, contexto para definir o tipo de query para `mx`. Então, como se vê abaixo, os emails seguem prioritariamente para o endereço “mail1.presidencia.pt”. Contudo, podem, de modo menos prioritário seguir para “mail2.presidencia.pt”. Estas prioridades são visíveis pelo valor que surge em primeiro lugar, antes dos endereços dos servidores de email responsáveis por receber os emails. Este número identifica apenas a prioridade. No caso de os valores serem iguais, a prioridade é a mesma. Tem mais prioridade o servidor cujo valor é maior.



```
core@xubuncore:~$ nslookup
> set query=mx
> presidencia.pt.
Server:      127.0.0.53
Address:     127.0.0.53#53

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

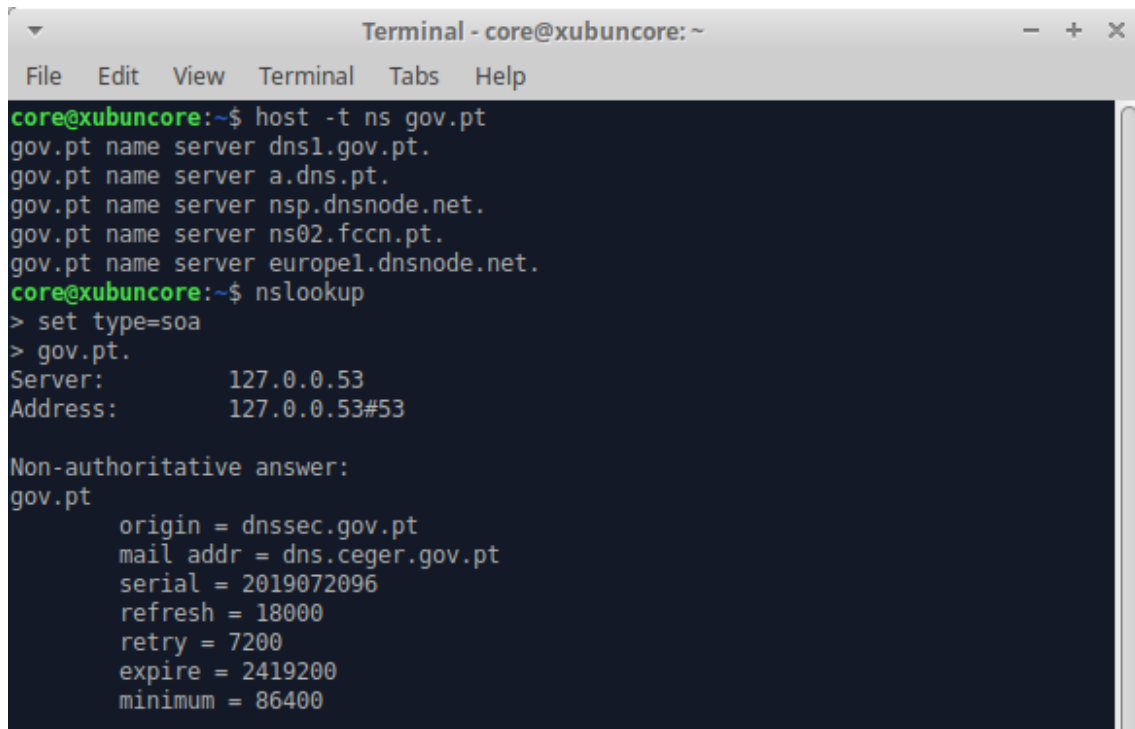
Figura 10: Informação obtida através da query MX

2.8 Alínea h)

| Que informação é possível obter, via DNS, acerca de gov.pt?

Via DNS, é possível extrair imensa informação sobre o servidor gov.pt., seguindo-se abaixo a respetiva lista com o conteúdo.

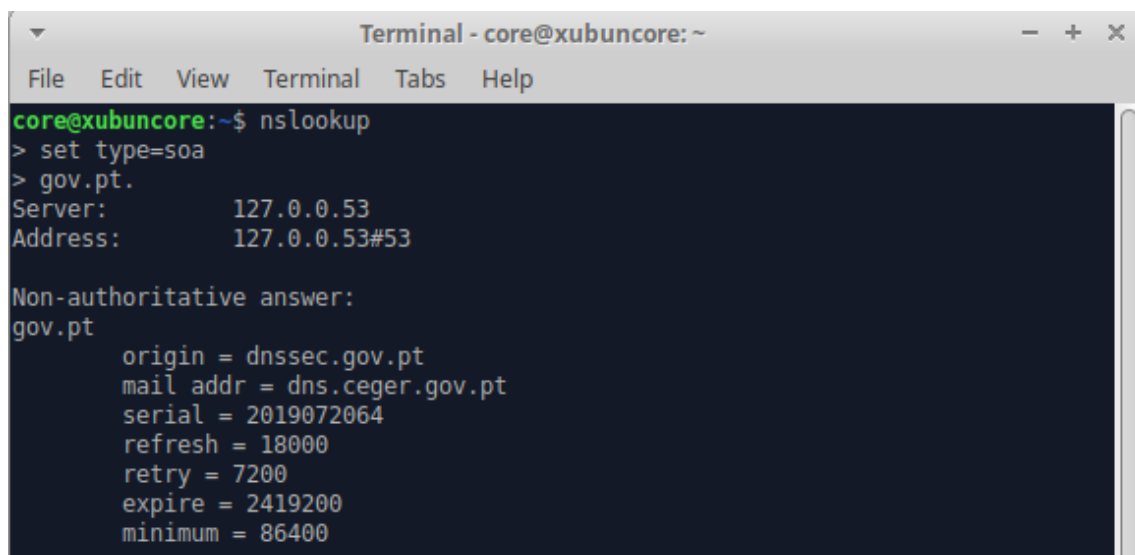
- Nome dos servidores associados a gov.pt;
- Informação autoritativa;
- Informação obtida com dig;



```
Terminal - core@xubuncore: ~
File Edit View Terminal Tabs Help
core@xubuncore:~$ host -t ns gov.pt
gov.pt name server dns1.gov.pt.
gov.pt name server a.dns.pt.
gov.pt name server nsp.dnsnode.net.
gov.pt name server ns02.fccn.pt.
gov.pt name server europel.dnsnode.net.
core@xubuncore:~$ nslookup
> set type=soa
> gov.pt.
Server:          127.0.0.53
Address:         127.0.0.53#53

Non-authoritative answer:
gov.pt
    origin = dnssec.gov.pt
    mail addr = dns.ceger.gov.pt
    serial = 2019072096
    refresh = 18000
    retry = 7200
    expire = 2419200
    minimum = 86400
```

Figura 11: Nome dos servidores associados a “gov.pt”



```
Terminal - core@xubuncore: ~
File Edit View Terminal Tabs Help
core@xubuncore:~$ nslookup
> set type=soa
> gov.pt.
Server:          127.0.0.53
Address:         127.0.0.53#53

Non-authoritative answer:
gov.pt
    origin = dnssec.gov.pt
    mail addr = dns.ceger.gov.pt
    serial = 2019072064
    refresh = 18000
    retry = 7200
    expire = 2419200
    minimum = 86400
```

Figura 12: Informação autoritativa sobre “gov.pt”

```
core@xubuncore:~$ dig gov.pt

; <<> DiG 9.16.1-Ubuntu <<> gov.pt
;; global options: +cmd
;; Got answer:
;; ->HEADER<<- opcode: QUERY, status: NOERROR, id: 18016
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;gov.pt.                                IN      A

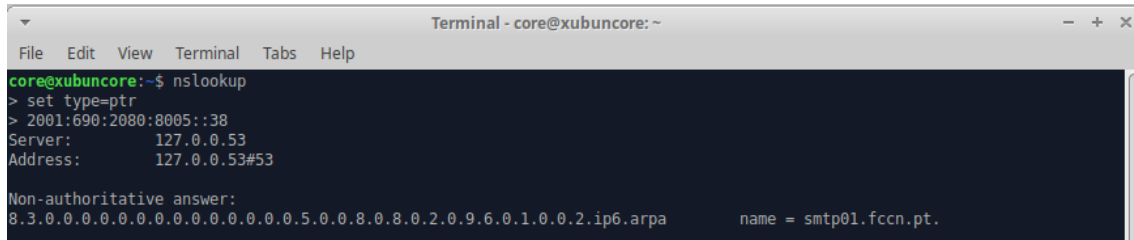
;; Query time: 23 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: dom nov 21 17:59:41 WET 2021
;; MSG SIZE rcvd: 35
```

Figura 13: Informação obtida com o comando dig

2.9 Alínea i)

Consegue interrogar o DNS sobre o endereço IPv6 2001:690:2080:8005::38 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?

Utilizando o `nslookup`, é possível percorrer a árvore do caminho inverso e descobrir qual o nome a que o endereço está associado. É possível obter o nome do domínio, neste caso, `smtp01.fccn.pt`. Se obtivéssemos algum problema, poderíamos obter a informação SOA, sabendo quem é o responsável pelo domínio e o seu endereço de email (mx).



```
Terminal - core@xubuncore: ~
File Edit View Terminal Tabs Help
core@xubuncore:~$ nslookup
> set type=ptr
> 2001:690:2080:8005::38
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
8.3.0.0.0.0.0.0.0.0.0.0.0.0.0.0.5.0.0.8.0.8.0.2.0.9.6.0.1.0.0.2.ip6.arpa      name = smtp01.fccn.pt.
```

Figura 14: Informação obtida com a query PTR

2.10 Alínea 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: uminho.pt).

2.10.1 Zona de Transferência

Esta é muitas vezes conhecida como *AXFR query type*, sendo um tipo de transação DNS (cliente-servidor) utilizada por administradores para replicar bases de dados em vários servidores DNS. Recorre também a TCP como transporte de dados e o processo de transferência de zona é sempre iniciado pelo cliente do servidor secundário a um servidor primário. A informação replicada é denominada de “zona”.

2.10.2 A operação de Transferência

Surge dividida em duas fases: a pré-fase ou preâmbulo, onde se determina se a transferência ocorre ou não, e a fase de transferência propriamente dita, onde os dados são replicados. Existem dois tipos de transferência, incremental ou completa, diferindo, inclusive, no tipo da query (AXFR ou IXFR).

- **Preâmbulo:** Compreende uma pesquisa do registro de recurso Start of Authority (SOA) para o “apex da zona”, o nó do namespace DNS que está no topo da “zona”. Um dos campos deste registro SOA, em particular o “número de série”, determina se a transferência de dados vai ocorrer. O cliente compara este número de série com o número de série da última cópia do registro SOA que ele possui. Se o número de série do registro transferido for maior, os dados na zona são considerados “alterados”(de alguma forma) e o secundário prossegue para solicitar a transferência de dados da zona real. Se os números de série forem idênticos, os dados na zona são considerados como não “alterados” e o cliente pode continuar a usar a cópia da base de dados que já possui, se existir.
- **Replicação:** A fase de transferência propriamente dita resume-se a:
 - CLIENTE: envia uma query do tipo AXFR por TCP;
 - SERVIDOR: responde com os registos de recurso para dada DNS na “zona” (a 1ª resposta é o registro SOA para o “apex da zona”);
 - Conexão termina com a repetição do registro SOA;

Podemos, então, aplicar um exemplo específico. Vamos tentar iniciar uma zona de transferência utilizando *uminho.pt*.. Seguem-se os respetivos passos:

1. Obter lista dos *name servers* do domínio *uminho.pt*;
2. Iniciar um pedido AXFR para obter uma cópia da zona do servidor primário.

As figuras seguintes representam estes mesmos passos, com dois comandos diferentes, **dig** e **host**.


```
Terminal - core@xubuncore: ~
File Edit View Terminal Tabs Help

core@xubuncore:~$ dig uminho.pt -t ns

; <<>> DiG 9.16.1-Ubuntu <<>> uminho.pt -t ns
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 9701
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;uminho.pt.                IN      NS

;; ANSWER SECTION:
uminho.pt.                3030    IN      NS      ns02.fccn.pt.
uminho.pt.                3030    IN      NS      dns.uminho.pt.
uminho.pt.                3030    IN      NS      dns3.uminho.pt.
uminho.pt.                3030    IN      NS      dns2.uminho.pt.

;; Query time: 0 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: dom nov 21 17:06:32 WET 2021
;; MSG SIZE rcvd: 118

core@xubuncore:~$ dig axfr uminho.pt @ns02.fccn.pt.

; <<>> DiG 9.16.1-Ubuntu <<>> axfr uminho.pt @ns02.fccn.pt.
;; global options: +cmd
;; Transfer failed.
```

Figura 15: Obtenção da informação com o comando dig

```
Terminal - core@xubuncore: ~
File Edit View Terminal Tabs Help

core@xubuncore:~$ host -t ns uminho.pt
uminho.pt name server ns02.fccn.pt.
uminho.pt name server dns.uminho.pt.
uminho.pt name server dns3.uminho.pt.
uminho.pt name server dns2.uminho.pt.
core@xubuncore:~$ host -l uminho.pt ns02.fccn.pt.
Using domain server:
Name: ns02.fccn.pt.
Address: 193.136.2.228#53
Aliases:

Host uminho.pt not found: 5(REFUSED)
; Transfer failed.
```

Figura 16: Obtenção da informação com o comando host

2.10.3 Vulnerabilidade e Prevenção

Como se verifica, em nenhum caso foi possível efetuar uma transferência de zona, isto porque o AXFR não oferece autenticação. Portanto, qualquer cliente pode solicitar a um servidor DNS uma cópia de toda a zona. Então, a menos que exista algum tipo de proteção, um invasor pode obter uma lista de todos os hosts de um domínio, o que pode possibilitar um ataque. Para evitar que esta vulnerabilidade ocorra, o servidor DNS deve ser configurado para permitir apenas transferências de zona de endereços IP confiáveis. No nosso caso, a transferência falha por isso mesmo, o nosso IP não está na lista de IPs confiáveis.

Segue-se, agora, para a **Parte II**.

3 Parte II

Pretende-se que crie um domínio CC.PT para a topologia de rede que estamos a usar nas aulas práticas (CC-Topo-2022.imn), de modo que se possam usar os nomes em vez dos endereços IP. No final deve, por exemplo, poder fazer-se “ping golfinho.cc.pt”, ou mesmo apenas “ping golfinho”, em vez de “ping 10.3.3.2”.

3.1 Modo de resolução

Esta segunda parte foi resolvida seguindo todos os passos descritos no enunciado. Cada secção especifica um passo específico e procura mostrar a prova de resolução. Então, neste contexto, instalamos, configuramos e testamos um domínio CC.PT, configurando um servidor DNS primário e secundário e executando-os no ambiente CORE.

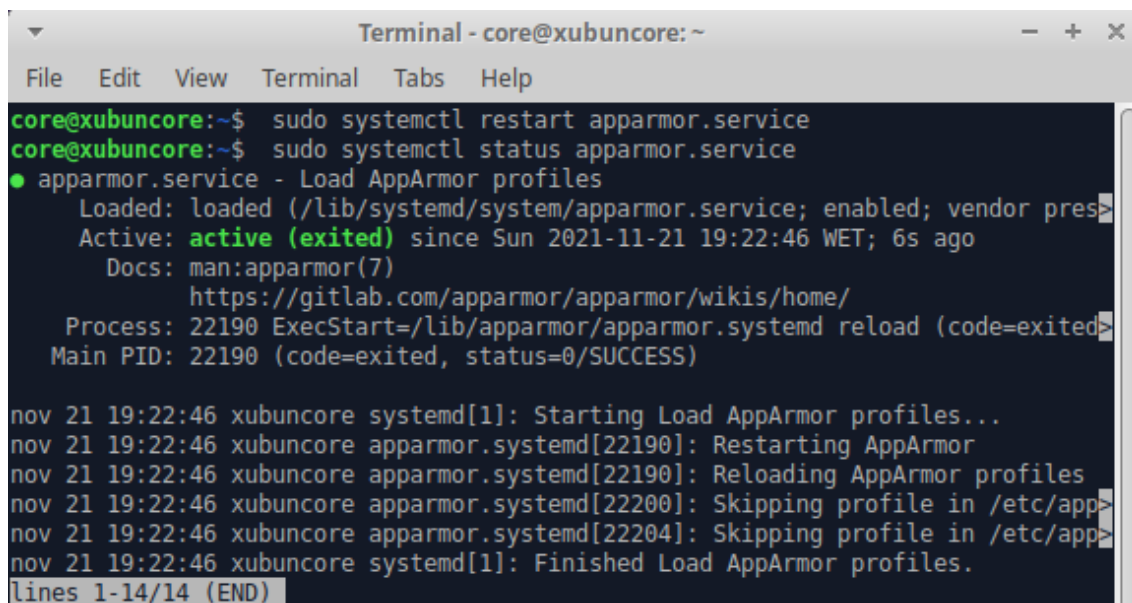
3.2 Preparativos especiais para o CORE

3.2.1 Passo 1) replicar ficheiros de configuração

Comçamos por copiar os ficheiros de configuração para uma nova pasta, já que todos os nós da topologia usam o mesmo filesystem. Isto permite que não se criem conflitos.

3.2.2 Passo 2) ver se o servidor DNS pré-instalado está em execução, parando-o de seguida se necessário

Passamos, depois à paragem do servidor DNS pré-instalado.

A terminal window titled "Terminal - core@xubuncore: ~" with a menu bar (File, Edit, View, Terminal, Tabs, Help). The terminal shows the following commands and output:

```
core@xubuncore:~$ sudo systemctl restart apparmor.service
core@xubuncore:~$ sudo systemctl status apparmor.service
● apparmor.service - Load AppArmor profiles
   Loaded: loaded (/lib/systemd/system/apparmor.service; enabled; vendor preset: enabled)
   Active: active (exited) since Sun 2021-11-21 19:22:46 WET; 6s ago
     Docs: man:apparmor(7)
           https://gitlab.com/apparmor/apparmor/wikis/home/
   Process: 22190 ExecStart=/lib/apparmor/apparmor.systemd reload (code=exited, status=0/SUCCESS)
   Main PID: 22190 (code=exited, status=0/SUCCESS)

nov 21 19:22:46 xubuncore systemd[1]: Starting Load AppArmor profiles...
nov 21 19:22:46 xubuncore apparmor.systemd[22190]: Restarting AppArmor
nov 21 19:22:46 xubuncore apparmor.systemd[22190]: Reloading AppArmor profiles
nov 21 19:22:46 xubuncore apparmor.systemd[22200]: Skipping profile in /etc/apparmor.d
nov 21 19:22:46 xubuncore apparmor.systemd[22204]: Skipping profile in /etc/apparmor.d
nov 21 19:22:46 xubuncore systemd[1]: Finished Load AppArmor profiles.
lines 1-14/14 (END)
```

Figura 17: Prova de reinício do servidor

3.2.3 Passo 3) reconfigurar apparmor para permitir que /usr/sbin/named aceda a ficheiros noutros locais

Procedeu-se à reconfiguração do apparmor de modo a que o ficheiro named aceda a ficheiros noutros locais, acrescentando-se as duas novas linhas de permissões.

```
core@xubuncore:~$ sudo cat /etc/apparmor.d/usr.sbin.named
[sudo] password for core:
# vim:syntax=apparmor
# Last Modified: Fri Jun  1 16:43:22 2007
#include <tunables/global>

/usr/sbin/named flags=(attach_disconnected) {
  #include <abstractions/base>
  #include <abstractions/nameservice>

  capability net_bind_service,
  capability setgid,
  capability setuid,
  capability sys_chroot,
  capability sys_resource,

  # /etc/bind should be read-only for bind
  # /var/lib/bind is for dynamically updated zone (and journal) files.
  # /var/cache/bind is for slave/stub data, since we're not the origin of it.
  # See /usr/share/doc/bind9/README.Debian.gz
  /etc/bind/** r,
  /var/lib/bind/** rw,
  /var/lib/bind/ rw,
  /var/cache/bind/** lrw,
  /var/cache/bind/ rw,
  /home/core/CC-TP3/primario/** r,
  /home/core/CC-TP3/secundario/** r,
```

Figura 18: Ficheiro usr.sbin.named com as respetivas permissões adicionadas.

Logo a seguir, reiniciamos o apparmor.

3.3 Configuração do servidor primário

Seguimos também um conjunto de passos que surgem descritos de seguida.

3.3.1 Editar o ficheiro `/etc/hosts`

Alterou-se este ficheiro de modo a que os servidores DNS se possam identificar.

```
127.0.0.1      localhost
127.0.1.1      xubuncore
10.2.2.1       Servidor1      ns.cc.pt
10.3.3.2       Golfinho       ns2.cc.pt
# The following lines are desirable for IPv6 capable hosts
::1           ip6-localhost ip6-loopback
fe00::0       ip6-localnet
ff00::0       ip6-mcastprefix
ff02::1       ip6-allnodes
ff02::2       ip6-allrouters
```

Figura 19: Ficheiro `/etc/hosts` alterado.

3.3.2 Editar o ficheiro `/CC-TP3/primario/named.conf.options`

Alterou-se este ficheiro para incluir os servidores do DI como forwarders.

```
options {
    directory "/var/cache/bind";

    // If there is a firewall between you and nameservers you want
    // to talk to, you may need to fix the firewall to allow multiple
    // ports to talk. See http://www.kb.cert.org/vuls/id/800113

    // If your ISP provided one or more IP addresses for stable
    // nameservers, you probably want to use them as forwarders.
    // Uncomment the following block, and insert the addresses replacing
    // the all-0's placeholder.

    forwarders {
        193.136.9.240;
        193.136.19.1;
    };

    //=====
    // If BIND logs error messages about the root key being expired,
    // you will need to update your keys. See https://www.isc.org/bind-keys
    //=====
    dnssec-validation auto;

    listen-on-v6 { any; };
};
```

Figura 20: Ficheiro `/CC-TP3/primario/named.conf.options` alterado.

3.3.3 Editar o ficheiro `/CC-TP3/primario/named.conf`

Alterou-se o ficheiro para incluir a indicação das novas zonas e substituir a diretoria.

```
// This is the primary configuration file for the BIND DNS server named.
//
// Please read /usr/share/doc/bind9/README.Debian.gz for information on the
// structure of BIND configuration files in Debian, *BEFORE* you customize
// this configuration file.
//
// If you are just adding zones, please do that in /etc/bind/named.conf.local

include "/home/core/CC-TP3/primario/named.conf.options";
include "/home/core/CC-TP3/primario/named.conf.local";
include "/home/core/CC-TP3/primario/named.conf.default-zones";

zone "cc.pt" {
    type master;
    file "/home/core/CC-TP3/primario/db.cc.pt";
    allow-transfer{
        10.3.3.2;
    };
};

zone "1.1.10.in-addr.arpa"{
    type master;
    file "/home/core/CC-TP3/primario/db.1-1-10.rev";
    allow-transfer{
        10.3.3.2;
    };
};

zone "2.2.10.in-addr.arpa"{
    type master;
    file "/home/core/CC-TP3/primario/db.2-2-10.rev";
    allow-transfer{
        10.3.3.2;
    };
};

zone "3.3.10.in-addr.arpa"{
    type master;
    file "/home/core/CC-TP3/primario/db.3-3-10.rev";
    allow-transfer{
        10.3.3.2;
    };
};
```

Figura 21: Ficheiro /CC-TP3/primario/named.conf alterado.

3.3.4 Criar o ficheiro de dados do domínio

Criou-se o ficheiro CC-TP3/primario/db.cc.pt.

```

$ORIGIN cc.pt.
$TTL 604800
@ IN SOA ns.cc.pt. g49pl04@cc.pt. (
    2 ; Serial
    604800 ; Refresh
    86400 ; Retry
    2419200 ; Expire
    604800 ) ; Negative Cache TTL

;Alias
ns IN CNAME Servidor1
ns2 IN CNAME Golfinho
www IN CNAME Servidor2
mail IN CNAME Servidor2
pop IN CNAME Servidor3
imap IN CNAME Servidor3

;LAN1
Portatil1 IN A 10.1.1.1
Portatil2 IN A 10.1.1.2
Portatil3 IN A 10.1.1.3
g49 IN CNAME portatil1

;LAN2
@ IN NS Servidor1
Servidor1 IN A 10.2.2.1
@ IN MX 5 Servidor2
Servidor2 IN A 10.2.2.2
@ IN MX 10 Servidor3
Servidor3 IN A 10.2.2.3

;LAN3
@ IN NS Golfinho
Golfinho IN A 10.3.3.2
@ IN NS Orca
Orca IN A 10.3.3.1
@ IN NS Foca
Foca IN A 10.3.3.3

;LAN4
Grilo IN A 10.4.4.1
Cigarra IN A 10.4.4.2
Vespa IN A 10.4.4.3

```

Figura 22: Ficheiro /CC-TP3/primario/db.cc.pt criado.

3.3.5 Criar os ficheiros reversos

Criou-se os ficheiros reversos em questão, como segue abaixo.

```

;
; BIND reverse data file for local loopback interface
;
$TTL 604800
@ IN SOA ns.cc.pt. g49pl04@cc.pt. (
    1 ; Serial
    604800 ; Refresh
    86400 ; Retry
    2419200 ; Expire
    604800 ) ; Negative Cache TTL

@ IN NS Servidor1.cc.pt.
@ IN NS Golfinho.cc.pt.

10.1.1.1 IN PTR Portatil1.cc.pt.
10.1.1.2 IN PTR Portatil2.cc.pt.
10.1.1.3 IN PTR Portatil3.cc.pt.

```

Figura 23: Ficheiro db.1-1-10.rev criado.

```

; BIND reverse data file for local loopback interface
;
$TTL      604800
@         IN      SOA      ns.cc.pt.      g49pl04@cc.pt. (
                        1          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )    ; Negative Cache TTL

@         IN      NS       Servidor1.cc.pt.
@         IN      NS       Golfinho.cc.pt.

10.2.2.1   IN      PTR      Servidor1.cc.pt.
10.2.2.2   IN      PTR      Servidor2.cc.pt.
10.2.2.3   IN      PTR      Servidor3.cc.pt.

```

Figura 24: Ficheiro db.2-2-10.rev criado.

```

;
; BIND reverse data file for local loopback interface
;
$TTL      604800
@         IN      SOA      ns.cc.pt.      g49pl04@cc.pt. (
                        1          ; Serial
                        604800     ; Refresh
                        86400      ; Retry
                        2419200    ; Expire
                        604800 )    ; Negative Cache TTL

@         IN      NS       Servidor1.cc.pt.
@         IN      NS       Golfinho.cc.pt.

10.3.3.1   IN      PTR      Golfinho.cc.pt.
10.3.3.2   IN      PTR      Orca.cc.pt.
10.3.3.3   IN      PTR      Foca.cc.pt.

```

Figura 25: Ficheiro db.3-3-10.rev criado.

3.3.6 Testar configurações e ficheiros de dados

Com os comandos do enunciado, verificou-se se os ficheiros tinham algum erro e em caso positivo, alterava-se o respetivo ficheiro.

```
core@xubuncore:~/CC-TP3$ /usr/sbin/named-checkconf -z /home/core/CC-TP3/primario/named.conf
zone localhost/IN: loaded serial 2
zone 127.in-addr.arpa/IN: loaded serial 1
zone 0.in-addr.arpa/IN: loaded serial 1
zone 255.in-addr.arpa/IN: loaded serial 1
zone cc.pt/IN: loaded serial 2
zone 2.2.10.in-addr.arpa/IN: loaded serial 1
core@xubuncore:~/CC-TP3$ /usr/sbin/named-checkzone cc.pt /home/core/CC-TP3/primario/db.cc.pt
zone cc.pt/IN: loaded serial 2
OK
core@xubuncore:~/CC-TP3$ /usr/sbin/named-checkzone 2.2.10.in-addr.arpa /home/core/CC-TP3/primario/db.2-2-10.rev
zone 2.2.10.in-addr.arpa/IN: loaded serial 1
OK
```

Figura 26: Teste de configuração e dos ficheiros

3.3.7 Executar o servidor

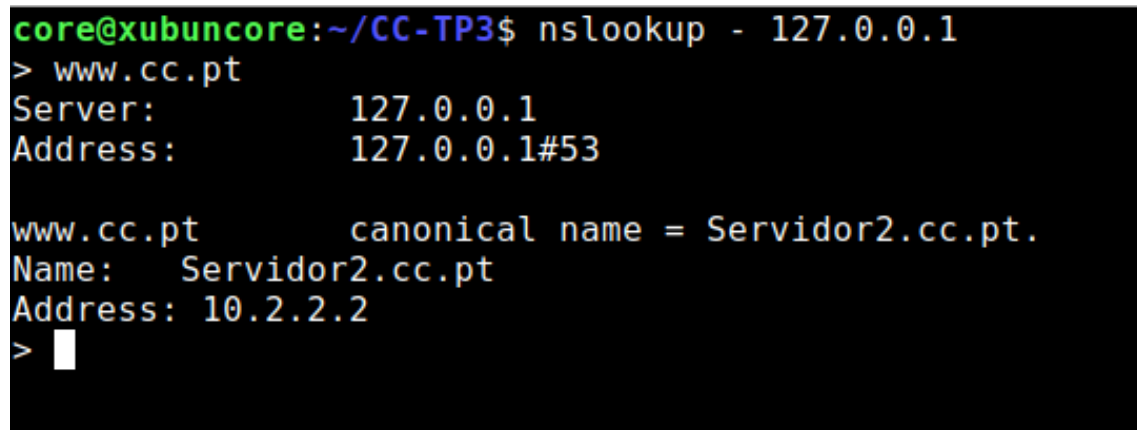
Para isso, paramos o serviço bind e reiniciamos o servidor.

```
22-Nov-2021 23:28:34.666 running
22-Nov-2021 23:28:34.670 zone 2.2.10.in-addr.arpa/IN: sending notifies (serial 1)
22-Nov-2021 23:28:34.670 zone 1.1.10.in-addr.arpa/IN: sending notifies (serial 1)
22-Nov-2021 23:28:34.670 zone 3.3.10.in-addr.arpa/IN: sending notifies (serial 1)
22-Nov-2021 23:28:34.670 zone cc.pt/IN: sending notifies (serial 2)
22-Nov-2021 23:28:34.670 network unreachable resolving './NS/IN': 2001:500:9f::42#53
22-Nov-2021 23:28:34.678 network unreachable resolving './NS/IN': 2001:500:12::d0d#53
22-Nov-2021 23:28:34.678 network unreachable resolving './NS/IN': 2001:dc3::35#53
22-Nov-2021 23:28:34.678 network unreachable resolving './NS/IN': 2001:7fe::53#53
22-Nov-2021 23:28:34.678 network unreachable resolving './NS/IN': 2001:7fd::1#53
22-Nov-2021 23:28:34.678 network unreachable resolving './NS/IN': 2001:500:2d::d#53
22-Nov-2021 23:28:34.678 network unreachable resolving './NS/IN': 2001:500:200::b#53
22-Nov-2021 23:28:34.678 network unreachable resolving './NS/IN': 2001:500:1::53#53
22-Nov-2021 23:28:34.678 network unreachable resolving './NS/IN': 2001:503:c27::2:30#53
22-Nov-2021 23:28:34.678 network unreachable resolving './NS/IN': 2001:500:a8::e#53
22-Nov-2021 23:28:34.678 network unreachable resolving './NS/IN': 2001:500:2f::f#53
22-Nov-2021 23:28:34.678 network unreachable resolving './NS/IN': 2001:500:2::c#53
22-Nov-2021 23:28:34.678 network unreachable resolving './NS/IN': 2001:503:ba3e::2:30#53
22-Nov-2021 23:28:35.870 timed out resolving './DNSKEY/IN': 193.136.19.1#53
22-Nov-2021 23:28:37.070 timed out resolving './DNSKEY/IN': 193.136.9.240#53
22-Nov-2021 23:28:37.070 network unreachable resolving './DNSKEY/IN': 2001:500:9f::42#53
22-Nov-2021 23:28:37.070 network unreachable resolving './DNSKEY/IN': 2001:500:12::d0d#53
22-Nov-2021 23:28:37.070 network unreachable resolving './DNSKEY/IN': 2001:dc3::35#53
22-Nov-2021 23:28:37.070 network unreachable resolving './DNSKEY/IN': 2001:7fe::53#53
22-Nov-2021 23:28:37.070 network unreachable resolving './DNSKEY/IN': 2001:7fd::1#53
22-Nov-2021 23:28:37.070 network unreachable resolving './DNSKEY/IN': 2001:500:2d::d#53
22-Nov-2021 23:28:37.070 network unreachable resolving './DNSKEY/IN': 2001:500:200::b#53
22-Nov-2021 23:28:37.070 network unreachable resolving './DNSKEY/IN': 2001:500:1::53#53
22-Nov-2021 23:28:37.070 network unreachable resolving './DNSKEY/IN': 2001:503:c27::2:30#53
22-Nov-2021 23:28:37.070 network unreachable resolving './DNSKEY/IN': 2001:500:a8::e#53
22-Nov-2021 23:28:37.070 network unreachable resolving './DNSKEY/IN': 2001:500:2f::f#53
22-Nov-2021 23:28:37.070 network unreachable resolving './DNSKEY/IN': 2001:500:2::c#53
22-Nov-2021 23:28:37.070 network unreachable resolving './DNSKEY/IN': 2001:503:ba3e::2:30#53
22-Nov-2021 23:28:44.670 resolver priming query complete
22-Nov-2021 23:28:44.670 managed-keys-zone: Unable to fetch DNSKEY set '.': timed out
22-Nov-2021 23:28:44.670 managed-keys.bind.jnl: open: permission denied
22-Nov-2021 23:28:44.670 managed-keys-zone: keyfetch_done:dns_journal_open -> unexpected error
22-Nov-2021 23:28:44.670 managed-keys-zone: error during managed-keys processing (unexpected error): DNSSEC validation may be at risk
```

Figura 27: Teste do servidor

3.4 Configuração do cliente e teste do primário

Primeiramente, temos de interrogar o servidor com o endereço 127.0.0.1 sobre o domínio www.cc.pt., isto é, interrogar o seu localhost.



```
core@xubuncore:~/CC-TP3$ nslookup - 127.0.0.1
> www.cc.pt
Server:          127.0.0.1
Address:         127.0.0.1#53

www.cc.pt        canonical name = Servidor2.cc.pt.
Name:   Servidor2.cc.pt
Address: 10.2.2.2
> 
```

Figura 28: Questionar servidor sobre o domínio www.cc.pt.

3.5 Configuração do servidor secundário

Segue-se o conjunto de passos na configuração do servidor secundário.

3.5.1 Editar o ficheiro named.conf.options

Alterou-se o ficheiro, da mesma forma que anteriormente, para incluir os servidores do DI como forwarders.

```
options {
    directory "/var/cache/bind";

    // If there is a firewall between you and nameservers you want
    // to talk to, you may need to fix the firewall to allow multiple
    // ports to talk. See http://www.kb.cert.org/vuls/id/800113

    // If your ISP provided one or more IP addresses for stable
    // nameservers, you probably want to use them as forwarders.
    // Uncomment the following block, and insert the addresses replacing
    // the all-0's placeholder.

    forwarders {
        0.0.0.0;
        193.136.9.240;
        193.136.19.1;
    };

    //=====
    // If BIND logs error messages about the root key being expired,
    // you will need to update your keys. See https://www.isc.org/bind-keys
    //=====
    dnssec-validation auto;

    listen-on-v6 { any; };
};
```

Figura 31: Ficheiro named.conf.options alterado.

3.5.2 Editar o ficheiro named.conf

Alterou-se o ficheiro para indicação das novas zonas. Contudo, este servidor é denominado de slave.

```
// This is the primary configuration file for the BIND DNS server named.
//
// Please read /usr/share/doc/bind9/README.Debian.gz for information on the
// structure of BIND configuration files in Debian, *BEFORE* you customize
// this configuration file.
//
// If you are just adding zones, please do that in /etc/bind/named.conf.local

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

zone "cc.pt" {
    type slave;
    file "/var/cache/bind/db.cc.pt";
    masters{
        10.2.2.1;
    };
};

zone "1.1.10.in-addr.arpa" {
    type slave;
    file "/var/cache/bind/db.1-1-10.rev";
    masters{
        10.2.2.1;
    };
};

zone "2.2.10.in-addr.arpa" {
    type slave;
    file "/var/cache/bind/db.2-2-10.rev";
    masters{
        10.2.2.1;
    };
};

zone "3.3.10.in-addr.arpa" {
    type slave;
    file "/var/cache/bind/db.3-3-10.rev";
    masters{
        10.2.2.1;
    };
};
};
```

Figura 32: Ficheiro named.conf alterado.

3.5.3 Teste dos ficheiros de configuração

Segue-se a prova da configuração correta dos ficheiros criados.

```
core@xubuncore:~/CC-TP3$ /usr/sbin/named-checkzone 1.1.10.in-addr.arpa /home/core/CC-TP3/primario/db.1-1-10.rev
zone 1.1.10.in-addr.arpa/IN: loaded serial 1
OK
core@xubuncore:~/CC-TP3$ /usr/sbin/named-checkzone 3.3.10.in-addr.arpa /home/core/CC-TP3/primario/db.3-3-10.rev
zone 3.3.10.in-addr.arpa/IN: loaded serial 1
OK
```

Figura 33: Testes realizados aos ficheiros criados.

3.5.4 Executar servidor e abrir a bash do nó Golfinho

```
22-Nov-2021 23:40:26.705 automatic empty zone: 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0
.0.0.0.0.0.0.0.0.0.0.0.0.0.0,IP6,ARPA
22-Nov-2021 23:40:26.705 automatic empty zone: D.F.IP6,ARPA
22-Nov-2021 23:40:26.705 automatic empty zone: 8.E.F.IP6,ARPA
22-Nov-2021 23:40:26.705 automatic empty zone: 9.E.F.IP6,ARPA
22-Nov-2021 23:40:26.705 automatic empty zone: A.E.F.IP6,ARPA
22-Nov-2021 23:40:26.705 automatic empty zone: B.E.F.IP6,ARPA
22-Nov-2021 23:40:26.705 automatic empty zone: 8.B.D.0.1.0.0.2.IP6,ARPA
22-Nov-2021 23:40:26.705 automatic empty zone: EMPTY,AS112,ARPA
22-Nov-2021 23:40:26.705 automatic empty zone: HOME,ARPA
22-Nov-2021 23:40:26.705 none:100: 'max-cache-size 90%' - setting to 1782MB (out
of 1980MB)
22-Nov-2021 23:40:26.705 configuring command channel from '/etc/bind/rndc.key'
22-Nov-2021 23:40:26.713 open: /etc/bind/rndc.key: permission denied
22-Nov-2021 23:40:26.713 couldn't add command channel 127.0.0.1#953: permission
denied
22-Nov-2021 23:40:26.713 configuring command channel from '/etc/bind/rndc.key'
22-Nov-2021 23:40:26.713 open: /etc/bind/rndc.key: permission denied
22-Nov-2021 23:40:26.713 couldn't add command channel ::1#953: permission denied
22-Nov-2021 23:40:26.713 not using config file logging statement for logging due
to -g option
22-Nov-2021 23:40:26.721 managed-keys-zone: loaded serial 90
22-Nov-2021 23:40:26.721 zone 0.in-addr.arpa/IN: loaded serial 1
22-Nov-2021 23:40:26.721 zone 127.in-addr.arpa/IN: loaded serial 1
22-Nov-2021 23:40:26.725 zone 255.in-addr.arpa/IN: loaded serial 1
22-Nov-2021 23:40:26.725 zone localhost/IN: loaded serial 2
22-Nov-2021 23:40:26.729 all zones loaded
22-Nov-2021 23:40:26.729 running
```

Figura 34: Testes realizados no nó Golfinho.

```
root@Portatil1:/tmp/pycore.40201/Portatil1.conf# nslookup - 10.3.3.2
> www.cc.pt
Server:          10.3.3.2
Address:         10.3.3.2#53

www.cc.pt        canonical name = Servidor2.cc.pt.
Name:   Servidor2.cc.pt
Address: 10.2.2.2
>
```

Figura 35: Testes lookup em qualquer nó da topologia.

4 Conclusões

Naturalmente, o trabalho realizado contribuiu para uma melhor compreensão da matéria lecionada nas aulas teóricas, alusiva ao DNS. Para além disso, este sistema hierárquico e distribuído mostrou a sua complexidade tanto a nível de ficheiros necessários para o seu funcionamento, como a nível de organização perante a busca de informação sobre um endereço.

Começou-se por analisar o ficheiro `resolv.conf`, que continha o conjunto dos parâmetros utilizados pelo resolver na sua busca. Ao longo das questões da primeira parte, procuramos realizar as perguntas através de vários comandos e não apenas com um, já que, diferentes comandos nos podem levar à mesma coisa, como acabamos por concluir.

Continuamos o trabalho com a criação dos dois servidores requisitados, seguindo todos os passos do enunciado: preparando o ambiente CORE (cópia de ficheiros e paragem do servidor DNS pré-instalado), configurando o servidor primário, o cliente e testando-o. Por fim, configuramos o servidor secundário. Todo este processo deu-nos um bom suporte sobre como todos os ficheiros estão relacionados entre si e como uns acedem a outros. Nota-se que não é fácil configurar estes servidores e que já há uma certa complexidade associada.