Departamento de Informática

RELATÓRIO DE ESTÁGIO

sobre

Construção de ferramentas de monitoragem e detecção de anomalias na rede PTEUnet

realizado no

Departamento de Informática da Faculdade de Ciências

e

PUUG – Grupo Português de Utilizadores de Sistemas Unix

por

Jorge Frazão de Oliveira

Lisboa, Julho de 1992

Departamento de Informática

RELATÓRIO DE ESTÁGIO

sobre

Construção de ferramentas de monitoragem e detecção de anomalias na rede PTEUnet

realizado no

Departamento de Informática da Faculdade de Ciências
e
PUUG – Grupo Português de Utilizadores de Sistemas Unix

por

Jorge Frazão de Oliveira

Responsável pelo DI/FCUL: José Legatheaux Martins
Responsável pelo PUUG: José Legatheaux Martins

Lisboa, Julho de 1992

Capítulo 1

Introdução

Este relatório pretende descrever o estágio que realizei no quinto ano da Licenciatura em Informática do Departamento de Informática da Faculdade de Ciências da Universidade de Lisboa (DI/FCUL). Teve como instituições de acolhimento: o PUUG (Grupo Português de Utilizadores de Sistemas Unix) e o DI/FCUL. O objectivo proposto inicialmente e cumprido no essencial foi a participação na montagem e construção de ferramentas de gestão da rede PTEUnet (rede internacional de serviços de aplicação dos sócios do PUUG).

O estágio desenrolou-se de Novembro de 91 a Julho de 92. Durante esse período contei com o auxílio importante do Prof. Legatheaux Martins (professor no DI/FCUL, membro da comissão executiva do PUUG e orientador do estágio) e do Eng. Henrique João Lopes Domingos (assistente no DI/FCUL e membro da comissão executiva do PUUG).

O local habitual de trabalho foi na sede do PUUG, para isso foi disponibilizada uma *workstation* NeXT (propriedade do PUUG) onde desenvolvi os vários projectos do estágio. Para além dessa máquina, tive acesso a outras máquinas do DI/FCUL e em particular ao *backbone* da PTEUnet, "dec4pt.puug.pt".

Nos capítulos que se seguem procederei à descrição do estágio.

- Capítulo 2 – Apresenta uma panorâmica geral do trabalho desenvolvido. Tem essencialmente como objectivo situar o estágio nas instituições de acolhimento (em particular no PUUG).

- Capítulo 3 – Dada a importância que sistema de designação, "Domain Name System" (DNS) assume no âmbito deste estágio a sua descrição é feita num único capítulo. Esta importância deve-se fundamentalmente às funções que assumi na administração de vários domínios e ao facto do projecto DDT (Domain Debug Tools) se basear neste sistema.

- Capítulo 4 – Apresenta de uma forma mais detalhada os vários projectos realizados durante o estágio. Descreve sucessivamente: a elaboração de

instrumentos de *accounting*, a construção de uma base de dados de aderentes da PTEUnet que permitisse automatizar as tarefas de administração do *backbone*, a experiência adquirida com o DNS, o desenvolvimento do DDT (uma breve introdução) e a análise preliminar para a montagem de um *archive server* (arquivo de ficheiros) na PTEUnet.

- Capítulo 5 – Dado o peso que o projecto DDT (Domain Debug Tools) assumiu, tanto em termos de tempo como de importância, em todo o trabalho realizado optei por o descrever num capítulo à parte.
- Apêndice A – Apresentação de um exemplo do relatório diário da PTEUnet, a que se faz referência no capítulo 4. Este relatório é elaborado pelos programas de *accounting*.
- Apêndice B – Lista dos atributos utilizados na definição de uma entidade na base de dados.
- Apêndice C – Sintaxe dos nomes em notação DNS.
- Apêndice D – Descrição de como um mensagem de correio electrónico é entregue, ou seja, quais os passos que seguidos pelo servidor antes de proceder à sua entrega. É importante perceber a semântica deste algoritmo, visto que uma incorrecta interpretação pode causar problemas no encaminhamento das mensagens.
- Apêndice E – Apresentação da tabela dos códigos ISO 3166 referidos no capítulo 5.

Capítulo 2

Panorâmica do trabalho desenvolvido

O estágio, DI-FCUL 2/3, teve como objectivo principal a participação na montagem e construção de ferramentas de gestão da rede de computadores PTE-Unet/EUnet (rede internacional de serviços do nível aplicação dos sócios do PUUG – Portuguese Unix User's Group). A proposta inicial compreendia as seguintes tarefas: participação na montagem e gestão dos servidores da rede, montagem do *archive server* e montagem do *bulletin board* – news Unix. Este trabalho estava subdividido em 4 fases:

1) Estudo dos diferentes serviços: DNS (sistema de designação), *mail router* (encaminhamento do correio electrónico), *news system* (sistema de teleconferência assistida por computador) e FTP (programa de transferência de ficheiros) [2 meses].

2) Construção de ferramentas de monitoragem e parametrização automática dos serviços a partir de uma base de dados [1 mês].

3) Construção de uma ferramenta de análise automática do estado do DNS nacional [1 mês].

4) Construção de uma interface de diálogo e gestão do *archive server* (arquivo de ficheiros) [2 meses].

Para além disto os estagiários deveriam participar na gestão da rede do Departamento de Informática da Faculdade de Ciências da Universidade de Lisboa (DI/FCUL). Inicialmente estava previsto serem alocados dois estagiários, contudo isso não se viria a verificar, acabando por ser só eu a participar no mesmo. O acolhimento foi repartido pelo PUUG e pelo departamento acima referido.

Antes de passar à descrição do trabalho realmente desenvolvido é importante fazer uma breve apresentação do que é o PUUG assim como das suas relações com as universidades e outras instituições de investigação.

O PUUG, Grupo Português de Utilizadores de Sistemas Unix, é uma associação sem fins lucrativos filiada na EurOpen (The European Forum for Open Systems). Tem a sua sede nas instalações do DI/FCUL. De entre os vários serviços que fornece aos seus sócios destaca-se um, quer pelo avultado investimento envolvido, quer pela sua importância. Estou a falar da PTEUnet, que está em funcionamento desde o segundo semestre de 91. Tal como o nome indica, trata-se de uma rede cooperativa de sistemas Unix[1] (PorTuguese Unix NETwork – ramo português da rede europeia EUnet) que oferece aos seus aderentes fundamentalmente três tipos de serviços: correio electrónico (*mail*), teleconferência assistida por computador (*news*) e conectividade mundial TCP/IP. A rede está organizada em torno de um nó central (o *backbone* "dec4pt.puug.pt.", propriedade do PUUG). Os nós membros da PTEUnet estão directamente ligados ao *backbone*. Este, por sua vez, está ligado a todos os outros *backbones* da Europa, permitindo o acesso aos 2600 nós dos mais de 20 países europeus membros da EUnet. O nó central da Europa, "mcsun.eu.net", pertencente à EurOpen, dá acesso às principais redes mundiais (UUnet, Internet, BITNET, etc). Devido aos elevados custos de comunicação envolvidos e ao ainda reduzido número de aderentes (20 até ao momento) a PTEUnet debate-se com problemas de viabilidade financeira. Em face disso estabeleceu, há uns meses a esta parte, um acordo com a FCCN que lhe permite utilizar a infra-estrutura da RCCN (Rede de Cálculo Científico Nacional) a preços universitários. Um outro apoio importante é o prestado pelo INESC (Instituto de Engenharia de Sistemas e Computadores) que garante a importação das *news* da Holanda. Como contrapartida o PUUG pôs à disposição de todas as universidades e instituições de investigação sem fins lucrativos, que já disponham de conectividade europeia, um serviço que garante o acesso ao Internet mundial e apoia a obtenção da autorização de acesso ao Internet dos EUA (NSFnet access). Sendo assim, podemos definir as relações entre PUUG, universidades, FCCN e INESC como uma relação de cooperação. Como prova disso podemos evidenciar os notáveis progressos realizados em Portugal nas infra-estruturas IP durante os últimos meses. Visto que passamos de uma fase de conectividade parcial, assegurada pelo INESC através de UUCP (Unix-to-Unix copy) sobre linha telefónica, para a fase de verdadeira conectividade ao Internet mundial. Desde o verão de 91 que passamos a estar ligados ao Internet mundial directamente por IP[10].

A minha colaboração com as 2 instituições: PUUG e DI/FCUL é anterior ao início do estágio. Com efeito, desde Janeiro de 91 que assumo as tarefas de monitor (na área dos PC's) no departamento. Com a PTEUnet comecei a colaborar alguns meses mais tarde, pouco tempo depois do seu aparecimento (no primeiro trimeste de 91). Inicialmente mais numa perspectiva de observador e

[1] Unix é uma marca registada da AT&T nos EUA e outros países.

de aprendizagem, visto que a minha experiência na área administração de sistemas era diminuta. Com o tempo fui adquirindo experiência e confiança o que me permitiu assumir um papel cada vez mais importante na administração da PTEUnet. Em paralelo com essa actividade a minha colaboração com o departamento foi-se alargando para as máquinas com o sistema Unix, principalmente devido à experiência que ia adquirindo nessa área.

Para além da monitoragem quotidiana, a minha colaboração estendeu-se à administração de vários utilitários, tais como: o BIND (servidor DNS), "sendmail", (servidor de encaminhamento de mensagens de *mail*) etc., à parametrização de modens, etc. A experiência adquirida na administração dos domínios (DNS) do PUUG, do departamento e de alguns dos aderentes da PTEUnet permitiu-me vir a colaborar mais tarde na montagem do *top-level domain* português em Portugal. Em paralelo com isso, instalei alguns utilitários ("sendmail"[1], "named", UUCP[17], vários utilitários da GNU[2], etc) tanto no departamento como na PTEUnet. É de destacar ainda a oportunidade que tive de instalar o sistema operativo em duas máquinas do departamento (tratou-se do *upgrade* do sistema operativo Ultrix 3.2 para Ultrix 4.2). Apesar de se tratarem de tarefas de certo modo repetitivas, elas assumiram um papel importante para um melhor conhecimento e à vontade sobre a administração do sistema.

Em meados de Outubro, depois de terminar o quarto ano da licenciatura, passei a dispôr de mais tempo, o que me permitiu colaborar com a PTEUnet a tempo inteiro e portanto a poder desenvolver trabalhos que requeriam uma investigação mais aprofunda. A partir de Novembro teve início o estágio propriamente dito. Assim desenvolvi no essencial três trabalhos antes de iniciar o projecto de maior peso:

- Análise de qual o método de *backups* para o nó central da PTEUnet, por outras palavras, responder a questões como: com que periodicidade devem ser feitos ?, que ficheiros deve incluir ?, etc [13, 18].

- Elaboração de *scripts*[3] que permitissem fazer o *accounting* dos aderentes, mas também dar uma ideia geral de qual o estado do nó central PTEUnet nas últimas 24 horas.

- Por fim, o maior dos três, tratou-se uma base de dados de aderentes que permitisse guardar informação técnica e administrativa dos aderentes e principalmente que servisse de base para a actualização das tabelas de configuração dos diversos utilitários, tais como o "sendmail", DNS, UUCP, etc. Para isso foi necessário fazer uma análise bastante pormenorizada do funcionamento dos diversos sub-sistemas o que possibilitou, sem margem para dúvidas, um melhor conhecimento do funcionamento dos mesmos.

[2] General Public License – denominação para uma classe de software que obedece a regras especiais de *copyright*. Estas regras permitem, entre outras coisas, a sua utilização sem o pagamento de taxa adicionais.

[3] Programas interpretados escritos na linguagem de "shell" e em "awk".

Finalmente, em finais de Janeiro iniciei um dos projectos de maior envergadura previstos inicialmente no planeamento do estágio e que acabaria por ser o único, devido à falta de tempo. Este investimento suplementar reflectiu-se principalmente na elaboração de manuais e de um artigo sobre o assunto, redigidos ambos em inglês. O projecto consistia em traços gerais na implementação de um conjunto de ferramentas que permitissem fazer análise e detecção de erros do DNS (ver Capítulo 5). Os objectivos foram atingidos estando actualmente este pacote a ser testado em diversos pontos da Europa e mesmo nos EUA. O planeamento do estágio incluia inicialmente um outro projecto que por falta de tempo acabou por ser adiado. Neste segundo projecto pretendia-se implementar um *archive server*, um pouco mais elaborado, com a noção de cache e se possível distribuído (ver Capítulo 4.4). Devido à falta de tempo não se passou de uma análise preliminar que espero poder vir a retomar depois de concluido o estágio.

Nos próximos capítulos vou tentar descrever cada um dos trabalhos citados acima, mais detalhadamente.

Capítulo 3

Domain Name System

3.1 Introdução

Ao nível da "suite" de protocolos TCP/IP a identificação de uma máquina no nível rede e transporte (isto é, o seu endereço de rede) é feita através de um inteiro de 32 bits, habitualmente designado endereço IP. Apesar de se tratar de um identificador válido a sua difícil memorização levou a que no nível aplicação as máquinas fossem designadas por um nome de mais alto nível.

Este capítulo pretende apresentar um sistema de designação de recursos, Domain Name System (DNS), que para além de associar os dois níveis de abstracção acima citados, gere ainda uma vasta gama de nomes associados a *hosts*, redes, famílias de protocolos, *mailboxes*, etc.

Evolução dos nomes

Inicialmente, quando o Internet possuia apenas algumas dezenas de máquinas e a frequência com que novas eram introduzidas era relativamente baixa, a associação entre os nomes e endereços IP era assegurada através de um ficheiro, "hosts.txt". Este ficheiro era gerido pelo NIC (Internet Network Information Center) e instalado em todos os sites manualmente (utilizando o FTP) no ficheiro "/etc/hosts".

Este modo de gestão centralizada veio demonstrar não ser adequado, principalmente devido à explosão do número de máquinas e à cada vez maior frequência com que apareciam novas máquinas. Em 1982 o ficheiro "hosts.txt" tinha cerca de 200 *hosts* e em 1988 já tinha mais de 5700.

Considerações de um Name System

Perante a inadequação deste tipo de gestão centralizada sentiu-se necessidade de desenvolver um sistema que a simplificasse. A elaboração deste sistema teve

como principais objectivos[11]:

- Permitir uma gestão local da base de dados, ou seja, possibilitar a cada organização a administração dos seus recursos localmente, independentemente das outras organizações.
- Ter um espaço de nomes hierarquizado com distribuição de autoridade, ou seja, a autoridade sobre um determinado ramo da árvore hierarquica pode ser delegada a uma organização, ficando esta responsável por todos os nomes pertencentes a esse ramo.
- Permitir uma extensibilidade na sua utilizações, quer relativamente aos atributos que guarda , quer às aplicações que o utilização.
- Oferecer um tamanho da base de dados virtualmente ilimitado.
- Permitir uma optimização das respostas utilizando, para isso, um mecanismo de *cache*.

Por outro lado assentou nos seguintes requisitos:

- O tamanho da base de dados é proporcional ao número de máquinas, no entanto, terá tendência a ser proporcional ao número de utilizadores.
- O tempo de vida da maioria da informação é longo, no entanto deve fornecer ferramentas que possibilitem uma rápida repercussão das alterações.
- Os clientes devem conseguir identificar falsos servidores.
- O acesso à informação é mais importante que a repercussão imediata das alterações ou que a garantia de consistência. O processo de distribuição das novas cópias é convergente e gerido através de um mecanismo de *time-outs*.
- Os servidores sempre que não conseguirem responder a uma pergunta podem tomar duas atitudes: perguntar a outro servidor e dar a resposta completa ou simplesmente dar ao cliente informação suficiente para que seja este a prosseguir com a pergunta. A última opção é preferível uma vez que liberta os servidores de alguma sobrecarga suplementar.

Componentes dos DNS

O Domain Name System (DNS) é um sistema de designação hierárquico e distribuído, idealizado para resolver o problema do crescente aumento de recursos. É hierárquico porque o espaço de nomes está dividido em sub-domínios. É distribuído porque a administração do espaço de nomes é delegada para o mais próximo possível dos recursos. Esta aproximação da administração aos recursos

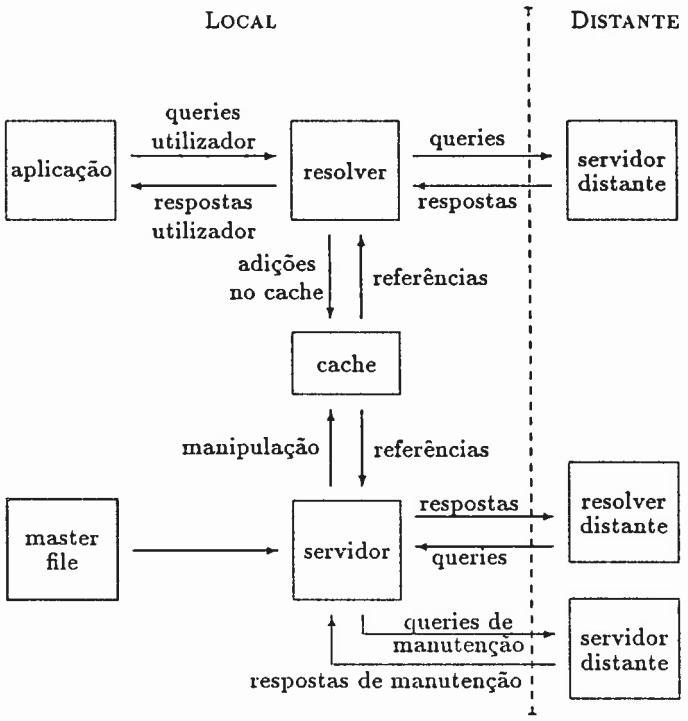

Figura 3.1: Configuração do sistema DNS

simplifica e optimiza o processo de introdução e alteração de informação na base de dados.

Este sistema é constituido por três grandes componentes:

DOMAIN NAME SPACE & RESOURCE RECORDS
: Específica a árvore estruturada de nomes e a informação associada aos nomes. A informação está associada aos nós e folhas da árvore sob o formato de Resource Records (RR).

NAME SERVERS
: São programas que gerem partes do espaço de nomes, designadas de zonas. O espaço de nomes está dividido por zonas de autoridade tendo os *name servers* o controlo completo dos nomes e RR pertencentes à(s) sua(s) zona(s). Um *name server* diz-se autoritário sobre todos os nomes pertencentes à zona de autoridade. Sempre que obtêm informação não per-

tencente à(s) sua(s) zona(s) autoritária(s), esta é colocada no *cache*, permitindo com isto melhorar a performance de futuras perguntas por essa informação.

RESOLVERS
: São programas que extraiem informação dos *name servers* para responder a solicitações de clientes.

A Figura 3.1[11] apresenta a configuração básica do DNS. As aplicações quando pretendem obter alguma informação solicitam ao *resolver*, sendo este que elabora a pergunta e a efectua ao(s) servidor(es), apresentando depois, a resposta à aplicação. O servidor interrogado pode não ser local, no entanto, se o for o seu cache é muitas vezes partilhado com o *resolver*. Entre os servidores existem dois tipos de interacções: quando um dos servidores assume o papel de *resolver* (cliente/servidor) e quando os servidores são autoritários sobre a mesma zona (servidor/servidor).

3.2 Domain Name Space & Resouce Records

Domain Name Space

Inicialmente no Internet era utilizado um *flat namespace* como política de nomes, ou seja, todos os nomes estavam ao mesmo nível. Para a gerir o NIC administrava uma base de dados centralizada[3].

Esta política tinha como grande vantagem o facto dos nomes serem consistentes e curtos. No entanto, com o constante aumento do número de nomes, verificou-se que a mesma não se adaptava à situação, uma vez que problemas como: o conflito de nomes face à cada vez menor hipótese de escolha, a dificuldade em administrar uma base de dados centralizada onde as alterações eram demasiado frequentes, etc. tornavam-se cada vez mais sérios. A solução passaria por hierarquizar a estrutura de nomes e descentralizar a sua administração[2].

Por isso o espaço de nomes DNS é uma árvore onde os nós representam nomes (ver Figura 3.2).

A árvore tem uma raiz "root". Um nome completo é identificado pelas componentes do caminho percorrido do nó com o nome em causa até à "root". Os componentes são palavras com o máximo de 63 bytes de comprimento. Sendo o comprimento máximo para um nome completo de 256 bytes. Por convenção um nome é escrito da direita para a esquerda com o "." a separar as diversas componentes. "root" é omitido do nome (ver Apêndice C).

Assim nomes como "puug.pt." ou "cc.purdue.edu." são nomes completos. Sempre que um nome não terminam com um ponto (nome relativo), a sua interpretação está dependente do nó da árvore onde for utilizado. Por exemplo o nome "fc.ul" só significa o domínio do DI/FCUL se for utilizado dentro do

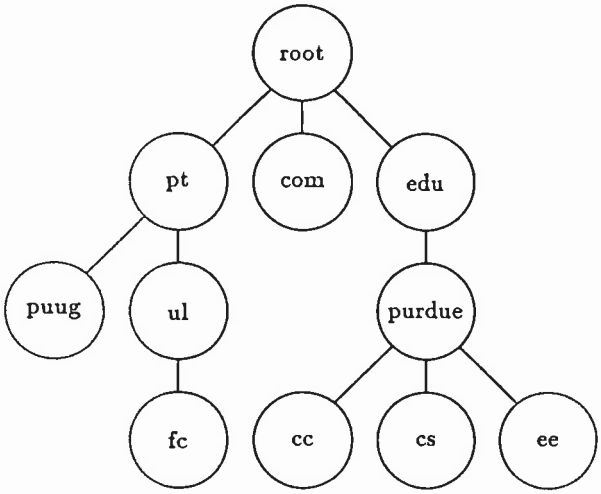

Figura 3.2: Exemplo do espaço de nomes DNS

domínio "pt.". Habitualmente os *resolvers* concatenam o domínio local aos nomes relativos criando assim um nome completo ("fc.ul.pt.").

Por vezes não se coloca o ponto no final de um nome completo ("fc.ul.pt"). Este comportamento não é errado no entanto se os *resolvers* utilizarem o método referido atrás pode provocar alguns problemas. Senão vejamos, supondo que no domínio "purdue.edu." alguém utilizava o nome "cs" com o significado de Checoslováquia, este seria contudo interpretado como "cs.purdue.edu.". Como alternativa muitos *resolvers* não prevêm a utilização de nomes relativos, ou seja, todos os nomes são interpretados como sendo completos independentemente de terem ou não o ponto à direita. Ambas as opções têm vantagens e inconvenientes.

Resource Records (RR)

Cada nó da árvore tem um conjunto de informação associada, podendo este ser vazio (apenas em situações de erro). Esta informação encontra-se organizada por RRs. Um RR é constituido por 6 componentes (Figura 3.3)[12]:

- OWNER – Identifica o nome (dono) a que a informação está associada.
- TYPE – Especifica o tipo do RR. Os tipos normalmente utilizados são:
  - A [Address]
  - NS [Name Server]

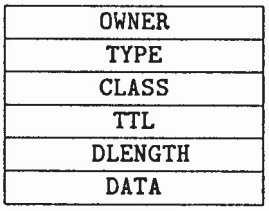

Figura 3.3: Componentes de um RR

  - CNAME [Canonical NAME]
  - SOA [Start Of Authority]
  - WKS [Well Known Service]
  - PTR [PoinTeR]
  - HINFO [Host INFOrmation]
  - MX [Mail eXchange]
  - TXT [TeXT]

- CLASS – Específica a família de protocolo para o tipo. Pode assumir os seguintes valores:
  - IN Sistema Internet
  - CH Sistema Chaos
  - HS Hesiod
- TTL (Time To Live) – Indica o tempo que o RR deve ser guardado no cache antes de ser deitado fora. Um valor nulo indica que o RR nunca deve ser colocado no cache.
- DLENGTH – Indica o tamanho do campo DATA, uma vez que este depende do tipo e classe especificados.
- DATA – Contém a informação relevante para o tipo e classe especificado. Os diversos conteúdos podem ser vistos mais abaixo.

Descrição de alguns RRs[9]:

- SOA (Start Of Authority)

```
<name> [<ttl>] [<class>] SOA <origin> <person> (
                         <serial>
                         <refresh>
                         <retry>
                         <expire>
                         <default TTL> )
```

Indica o início da zona de autoridade. A zona termina no próximo SOA RR.
`<origin>` – nome do *host* onde o *master file* reside.
`<person>` – *mailbox* da pessoa responsável pela zona.
`<serial>` – número de versão da zona. Deve ser incrementado sempre que for feita alguma alteração.
`<refresh>` – intervalo de tempo (em segundos) que o secundário espera para verificar com o primário se a versão foi alterada.
`<retry>` – intervalo de tempo (em segundos) que o secundário espera antes de tentar contactar o primário depois de ter falhado a última tentativa.
`<expire>` – tempo máximo (em segundos) que o secundário disponibiliza a informação da zona sem contactar o primário.
`<default TTL>` – TTL mínimo para todos os RRs da zona. É aconselhável que algum tempo antes de se fazerem alterações, principalmente se estas implicarem grandes mudanças, que estes valores sejam reduzidos para que as inconsistências predurem no sistema o mínimo de tempo possível.

- NS (Name Server)

  ```
  <domain> [<ttl>] [<class>] NS <server>
  ```

  Indica que a máquina `<server>` é autoritária sobre o domínio.

- A (Address)

  ```
  <host> [<ttl>] [<class>] A <address>
  ```

  É utilizado para associar um endereço a um nome. No caso do Internet, o `<address>` é um inteiro de 32 bits (endereço IP).

- CNAME (Canonical NAME)

  ```
  <nickname> [<ttl>] [<class>] CNAME <host>
  ```

  É utilizado para associar um *host* a um alias (`<nickname>`).

- HINFO (Host INFOrmation)

  ```
  <host> [<ttl>] [<class>] HINFO <hardware> <software>
  ```

  Apresenta informação sobre um dado *host*. Essa informação é constituida por duas partes: descrição do hardware e descrição do software.

- WKS (Well Known Service)

  ```
  <host> [<ttl>] [<class>] WKS <address> <protocol> <service>
  ```

Indica quais os serviços que `<host>` disponibiliza. Os serviços estão associados a um endereço (`<address>`) e a um protocolo (`<protocol>`). O campo `<service>` é um *bitmap* que indica quais os serviços disponibilizados.

- MX (Mail eXchange)

  `<domain> [<ttl>] [<class>] MX <preference> <host>`

  Especifica quais as máquinas que devem receber o mail para para `<domain>`. O campo `<preference>` indica qual a prioridade com que esta informação deve ser utilizada, ou seja, quanto menor for o valor, maior é a prioridade de este `<host>` receber essas mensagens (Ver Apêndice D).

- PTR (PoinTeR)

  `<special-name> [<ttl>] [<class>] PTR <name>`

  É utilizado como apontador para um outro nó da árvore. Habitualmente é utilizado no *reverse mapping*, ou seja, para permitir obter o *host* com determinado endereço IP. Neste caso o campo `<special-name>` tem um nome do tipo: `XXX.YYY.ZZZ.WWW`.in-addr.arpa. onde `WWW.ZZZ.YYY.XXX` é um endereço IP.

A definição dos RRs é feita num ficheiro de texto, *master file*. Sendo este carregado para memória pelo primário da zona. Os RRs são definidos um por linha (sendo possível definir em mais do que uma linha utilizando os parênteses) segundo as seguintes regras:

1. Sempre que uma linha começa por um espaço em branco o OWNER é o mesmo do RR anterior.

2. A seguir são especificados os campos: TTL, TYPE e CLASS. Como os três conjuntos são disjuntos entre si são muitas vezes omitidos os campos: TTL e/ou CLASS.

3. Finalmente é definido o campo DATA.

A Tabela 3.1 apresenta um pequeno exemplo em um *master file*.

O primeiro RR define o começo de uma zona de autoridade, neste caso da zona "puug.pt". Os NS RRs indicam quais os servidores autoritários sobre a zona. Neste caso, um deles está contido na própria zona, "dec4pt.puug.pt" e os restantes dois são exteriores. O *mail* dirigido para "puug.pt" deve ser entregue a uma das máquinas: "dec4pc.puug.pt" ou "inesc.inesc.pt" mas preferencialmente à primeira. Existem ainda mais dois RR associados ao nome "puug.pt", trata-se de TXT RRs utilizados apenas para associar informação administrativa a esse nome. Como se pode observar pelos A RRs existem dois *hosts* nesta zona: "dec4pt.puug.pt" e "colombo.puug.pt". O primeiro é um

```
puug.pt.     SOA    dec4pt.puug.pt. jalm.puug.pt. (
                    199204100 ;serial
                    28800     ;refresh
                    7200      ;retry
                    604800    ;expire
                    86400 )   ;minim
             NS     dec4pt.puug.pt.
             NS     inesc.inesc.pt.
             NS     tristan.fct.unl.pt.
             TXT    "Portuguese Unix network"
             TXT    "Secretariat: sec@puug.pt"
             MX     10 dec4pt.puug.pt.
             MX     20 inesc.inesc.pt.
dec4pt       A      192.84.62.1
             HINFO  DECsystem-3100 Ultrix-4.0
             WKS    192.84.62.1 tcp smtp domain
             WKS    192.84.62.1 udp domain
colombo      A      192.67.76.4
             MX     20 dec4pt.puug.pt.
col          CNAME  colombo.puug.pt.
```

Tabela 3.1: Exemplo de um *master file*

DECsystem 3100 a correr Ultrix 4.0 (HINFO RR) e disponibiliza os serviços "smtp" e "domain" (DNS) sobre "tcp" e o "domain" sobre "udp" (WKS RR) . Relativamente ao segundo, podemos ainda verificar que este tem um alias, "col.puug.pt", e que todo o *mail* dirigido a "user@colombo.puug.pt" será entregue ao *host* "dec4pt.puug.pt" que depois o encaminhará para ele.

Queries

As mensagens enviadas aos servidores para obter determinada informação são designadas por *queries* (perguntas). No Internet podem ser feitas utilizando UDP (Internet User Datagram Protocol) ou TCP (Internet Transmission Control Protocol).

O DNS suporta dois tipos de queries: recursivas e iterativas. No primeiro caso o servidor contacta outros servidores sempre que não consegue resolver a pergunta sozinho. Enquanto no segundo caso, se não for capaz de responder sozinho entrega ao cliente a informação suficiente (apontadores para os servidores autoritários ou responsáveis pela delegação de autoridade da zona em causa) para que seja este a continuar a *query*. As *queries* iterativas são preferíveis uma vez que libertam os servidores de algumas operações que podem ser implementadas pelos *resolvers*. Existem mesmo servidores que não aceitam *queries* recursivas, nomeadamente, os servidores dos domínio mais elevados da árvore

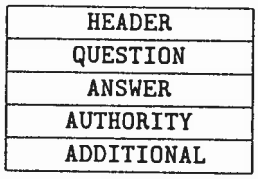

Figura 3.4: Componentes de uma *query* DNS

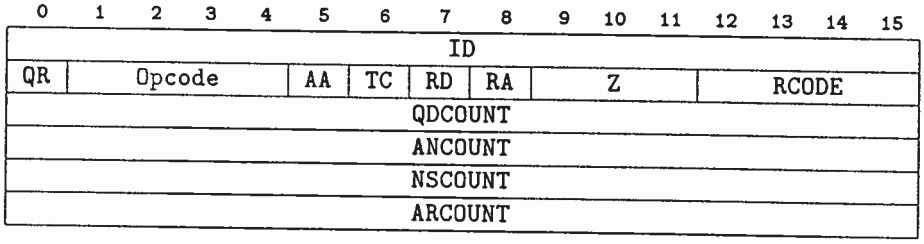

Figura 3.5: Componentes do *header* de uma *query* DNS

("root", *top-level domains*, etc).

Uma mensagem deste tipo é constituida por 5 secções (ver Figura 3.4):

- HEADER – Especifica quais as restantes secções presentes, tipo de mensagem, etc (ver Figura 3.5)[12].
  - ID, identificador da mensagem, atribuído pelo programa que a cria.
  - QR, especifica se se trata de uma pergunta ou uma resposta.
  - Opcode, especifica o tipo de *query*: standard ou inversa.
  - AA [Authoritative Answer], indica se a resposta é ou não autoritária.
  - TC [TrunCation], indica se a resposta está ou não completa.
  - RD [Recursion Desired], indica se o servidor deve, caso necessite, prosseguir a *query* recursivamente.
  - RA [Recursion Available], indica se o servidor pode ou não executar uma *query* recursiva.
  - Z, reservado para uso futuro.
  - RCODE [Response Code], indica o tipo de erro que ocorreu.
  - QDCOUNT, indica o número de RRs incluidos na secção QUESTION.

  - ANCOUNT, indica o número de RRs incluidos na secção ANSWER.
  - NSCOUNT, indica o número de RRs incluidos na secção AUTHORITY.
  - ARCOUNT, indica o número de RRs incluidos na secção ADDITIONAL.

- QUESTION – Indica qual o nome, tipo e classe sobre a qual inside a pergunta. É constituida por três campos:
  - QNAME, nome de que se pretende obter informação.
  - QTYPE, pode ser um de:
    * "AXFR" – transferência de zona;
    * "*" – todos os RRs;
    * TYPE – tipo de RR indicado.
  - QCLASS, pode ser um de:
    * "*" – todas as classes;
    * CLASS – classe indicada.

- ANSWER – Contém o(s) RR(s) que verificam a *query*.

- AUTHORITY – Contém RRs que apresentam os servidores autoritários (lista de NS RRs). Normalmente são incluidos se o servidor em causa não conhecer a resposta (supondo que não suporta queries recursivas) mas tem conhecimento dos servidores a quem foi delegada a autoridade sobre o nome.

- ADDITIONAL – Contém RRs que, de uma maneira ou de outra, poderão vir a ser úteis aquando da utilização dos RRs incluídos nas outras secções. É o caso de uma pergunta pelos MX RRs para um determinado domínio, junto com estes (caso existam) são adicionados os A RRs correspondentes. Com isto pretende-se poupar uma *query*, visto que com grande probabilidade a *query* seguinte seria a perguntar pelos endereços dos servidores obtidos.

As *queries* ainda se podem classificar em: standard e inversa. O primeiro consiste em dado um nome, um tipo e uma classe obter os RRs que os verificam. No segundo caso, tal como o nome indica, esta é feita no sentido inverso, ou seja, dado por exemplo um endereço IP obter o *host* que lhe está associado. Este tipo de *queries* nem sempre consegue obter a resposta correcta, visto que a base de dados não está organizada por endereços IP ou qualquer outro recurso. Talvez por essa razão só o primeiro tipo (endereço IP) é actualmente utilizado.

Com o objectivo de diminuir o tamanho das mensagens é utilizado um algoritmo de compressão que consiste, essencialmente, em eliminar todas as

HEADER	Opcode=SQUERY, RESPONSE, AA
QUESTION	QNAME=puug.pt., QCLASS=IN, QTYPE=MX
ANSWER	puug.pt. 86400 MX 10 dec4pt.puug.pt.
AUTHORITY	
ADDITIONAL	dec4pt.puug.pt. 86400 A 192.84.62.1

HEADER	Opcode=SQUERY, RESPONSE, NORECURSIVE
QUESTION	QNAME=puug.pt., QCLASS=IN, QTYPE=A
ANSWER	
AUTHORITY	puug.pt. .345600 NS dec4pt.puug.pt.
	puug.pt. 345600 NS inesc.inesc.pt.
	puug.pt. 345600 NS trinstan.fct.unl.pt.
ADDITIONAL	dec4pt.puug.pt. 346500 A 192.84.62.1
	inesc.inesc.pt. 345600 A 146.193.0.1
	trinstan.fct.unl.pt. 345600 A 192.68.178.190

Figura 3.6: Exemplos de *queries* DNS e respectivas respostas

repetições de nomes, substituindo-os por apontadores para a primeira ocorrência.

A Figura 3.6 apresenta dois exemplos de *queries* DNS. No primeiro exemplo pretende-se obter os MX RRs associados ao domínio "puug.pt.". A resposta apresenta um MX RR que aponta para o *host* "dec4pt.puug.pt." com uma preferência de 10. Junto com este RR vem a informação necessária para que se possa contactar o *host* em causa, ou seja, o seu endereço IP. No segundo caso pretende-se obter os endereços IP associados ao nome "puug.pt.". Como o servidor interrogado não tinha essa informação e a *query* não era recursiva, este limitou-se a dar a lista se servidores autoritários sobre o nome para que o cliente pudesse prosseguir a *query*. Junto com a lista de servidores adicionou os respectivos endereços, visto que seriam necessários para a continuação da *query*.

3.3 Name Servers

Name servers (servidores) são programas que gerem partes do espaço de nomes DNS. A estas partes é usual chamar-se zonas[4]. Os servidores dizem-se autoritários sobre toda a informação contida nas suas zonas. Uma zona deve ser gerida por mais do que um servidor garantindo com isto a redundância e acessibilidade dos seus dados. Por outro lado, cada servidor pode gerir em simultâneo mais do que uma zona.

A sua principal função é responder a *queries* feitas na maioria dos casos pelos *resolvers*. No entanto, para que isso seja possível é necessário que exista uma intercomunicação entre os servidores, nomeadamente para a manutenção das cópias

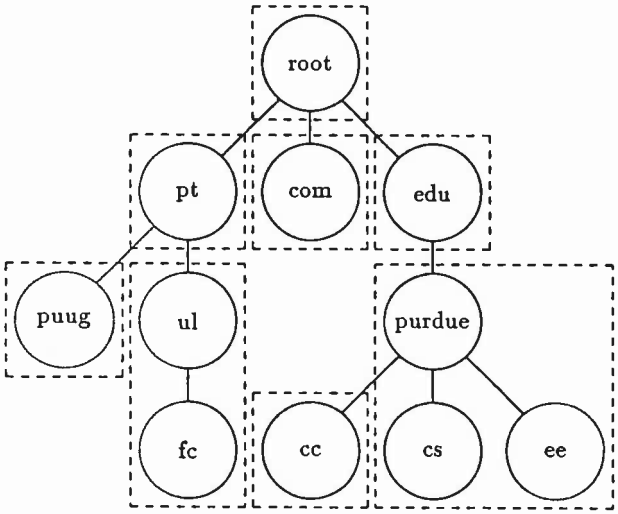

Figura 3.7: Exemplo de zonas de autoridade

Para além da informação autoritária os servidores implementam um mecanismo de *cache* que lhes permite guardar temporariamente informação não autoritária que conseguiram obter respondendo a *queries*. Para isso cada RR tem um campo, TTL, que indica o tempo que este pode ser guardado no *cache*. Este mecanismo permite melhorar a eficácia da utilização do *cache*.

Definição de zonas

A base de dados DNS está particionada de duas maneiras: por classes e por "cortes" feitos entre os nós da árvore de nomes.

A divisão por classes é relativamente simples, uma vez que diferentes classes são organizadas, delegadas e administradas separadamente pelos servidores.

A divisão por "corte" pode resumir-se na separação da árvore por zonas de autoridade.

As zonas são normalmente identificadas pelo nome do SOA, ou seja, o nó mais alto da árvore que lhe pertence.

Cada zona inclui pelo menos um nó da árvore. Como se pode ver na Figura 3.7 a zona "puug.pt" é constituída só por um nó enquanto a zona "ul.pt" inclui dois nós.

Este tipo de subdivisão da árvore permite que as organizações administrem a sua zona separadamente (alterem dados, criem mais nós, deleguem a autoridade de um sub-domínio, etc) o que simplifica efectivamente essa tarefa, visto

que a quantidade de informação é mais reduzida e por outro lado existe uma aproximação da administração aos recursos.

Uma zona é definida por quatro tipos de informação[11]:

1. Informação associada aos nós pertencentes à zona. Normalmente designada por informação autoritária. Encontra-se codificada em RRs.

2. Informação que define o nó superior da zona. Assume um papel importante na sua administração, uma vez que define vários dados administrativos, como: intervalos de tempo entre dois *updates*, lista de servidores autoritários, etc. Nela estão incluidos a lista de NS RRs e o SOA RR associados à zona.

3. Informação que descreve as subzonas delegadas. Para além da definição dos limites inferiores é também necessário definir até onde essa autoridade se estende, ou seja, definir quais os sub-domínios para que foi delegada a autoridade. Neste grupo estão incluidos as listas de NS RRs e os SOA RRs das subzonas.

4. Informação que permite o acesso aos servidores das subzonas, também designada por *glue data*. Uma das funções da zona é disponibilizar a informação necessária para a comunicação com os servidores das suas subzonas. A lista de NS RRs não é suficiente, uma vez que para contactar os servidores aí indicados, é necessário o seu endereço IP. Em particular, se o servidor estiver dentro da subzona de que é autoritário nunca se conseguirá obter o seu A RR visto que não se consegue contactar o servidor responsável por essa informação (ou seja, ele próprio). Por outras palavras, está criado um *deadlock* que só é resolvido introduzindo na zona "pai" (zona que delegou a autoridade) os A RRs dos servidores que estejam nessas condições. Note-se que esta informação só é necessária se o servidor pertencer a um subzona.

   Aos A RRs de servidores da zona ou de uma subzona que não pertençam à zona é usual chamar-se *glue records* dado o contributo que prestam para ligarem as duas zonas.

Funcionamento dos name servers

Os servidores estão ligados entre si por apontadores. Quando um servidor delega a autoridade mantem apontadores para os servidores que passaram a gerir a subzona. Nesta caso os apontadores são os NS RRs. Seguindo esses apontadores é possível alcançar os servidores autoritários sobre um determinado nome partindo dos servidores de "root". Quando um servidor da "root" é interrogado com uma determinada *query* pode tomar um de dois comportamentos possíveis, dependendo do facto de este ser autoritário sobre essa informação ou ter delegado a responsabilidade a outros (situação mais frequente). No primeiro caso

responde imediatamente à *query* enquanto no segundo devolve os apontadores (lista de NS RRs) para os servidores agora responsáveis por essa informação. Para simplificar estamos a supor que o servidor em causa não possuia a informação pretendida em *cache* assim como não aceitava *queries* recursivas.

Através deste método é sempre possível atingir os servidores autoritários sobre qualquer nome completo. Por outro lado, este processo de ligação entre os servidores permite que a estrutura da árvore seja totalmente independente da topologia fisica da rede (Internet).

Supondo que alguém nos EUA pretendia enviar uma mensagem de *mail* para "user@puug.pt". O seu *resolver* ao tentar obter os MX RRs associados ao domínio "puug.pt" começaria por perguntar a um servidor local que lhe indicaria os servidores de "root" para prosseguir a *query*. Depois de executar a mesma *query* a um deles obteria a lista de servidores autoritário sobre a zona "pt.", uma vez que lhes tinha sido delegada a autoridade para a sub-árvore "pt.". O processo prosseguia até obter a lista de servidores responsáveis pelo domínio "puug.pt". Só depois de interrogado um desses servidores obteria a informação pretendida (caso existisse). Mais uma vez estamos a supor que as *queries* eram iterativas, que nenhum dos servidores interrogados era simultaneamente autoritário sobre a zona e subzona e que essa informação também não estava disponível no *cache* de nenhum desses servidores.

Para garantir a acessibilidade da informação, uma zona deve ser gerida por pelo menos dois servidores. Sempre que possível estes devem estar afastados fisicamente para minimizar o impacto de falhas na rede, *gateways*, etc. Esta duplicação de informação é mantida segundo um esquema de *master/slave* ou primário/secundário. Todas as alterações são feitas directamente na base de dados do primário. Periodicamente, os secundários contactam o primário para o indagarem do número de versão. Se o número que o primário tiver for superior (e só superior) é feita a transferência da zona. A este processo é usual chamar-se *zone transfer*. Os intervalos de tempo com que os secundários contactam o primário são parametrizaveis por valores especificados no SOA RR. A única distinção entre primário e secundários é sentida na altura do *updates*, uma vez que os secundários podem ter versões desactualizadas durante períodos de tempo imediatemente a seguir a uma alteração. Em tudo o resto é indiferente qual dos servidores contactar, uma vez que ambos são autoritários sobre a mesma informação.

Uma opção ainda não implementada e que traria grandes vantagens é a introdução de TTLs negativos. Sempre que era obtida uma resposta negativa os dados que a tinham originado seriam colocados no *cache*, mas com um valor negativo para que em posteriores insistências na mesma pergunta fosse dado de imediato uma resposta negativa. Em suma, um comportamento simétrico ao tomado com as respostas positivas e que permitiria, à semelhança deste, optimizar os tempos de resposta.

3.4 Resolvers

Resolvers são programas que interrogam os servidores em nome dos clientes (em geral aplicações). Para os contactarem as aplicações ("sendmail", FTP, TELNET, etc.) utilizam sub-rotinas ou *library calls* disponibilizados pelo sistema. Normalmente o *resolver* encontra-se na mesma máquina que o cliente e contacta servidores, também em geral, situados em máquinas distantes (Figura 4.1). Os seus tempos de resposta podem variar de milisegundos a alguns segundos, dependendo se a informação pretendida estava em *cache* local ou se foi necessário contactar alguns servidores. Como existe uma convergência das perguntas dos diversos processos para um ponto de acesso (*resolver*), permite que muitas das respostas já estejam no *cache* local quando forem solicitadas. Isto permite melhorar a sua eficiência, conseguindo-se muitas vezes eliminar os atrasos provocados pela rede ou pela sobrecarga dos servidores. Por outro lado, também desempenham um papel importante na libertação de algumas tarefas aos servidores, nomeadamente as relativas às perguntas recursivas. Nas perguntas iterativas são os *resolvers* que implementam todo o processo de seguir os apontadores até atingir um dos servidores autoritários. Enquanto aos servidores é exigido apenas, que respondam com base na informação que tiverem disponível no momento.

Sempre que exista um servidor local o seu *cache* deve ser partilhado com o *resolver*. Com isso consegue-se um *cache* mais rico e evitam-se muitas das perguntas ao servidor local.

Capítulo 4

Definição detalhada do trabalho desenvolvido

Este capítulo descreve o trabalho desenvolvido durante o estágio. Tratarei sucessivamente: a elaboração de instrumentos de *accounting*, a elaboração de uma base de dados de aderentes da PTEUnet, a experiência adquirida com o sistema DNS, a elaboração do pacote Domain Debug Tools e a análise da montagem de um arquivo de ficheiros da PTEUnet.

4.1 Instrumentos de accounting

A administração de uma máquina como o nó central da PTEUnet engloba várias tarefas. Estas dividem-se fundamentalmente em dois grupos: monitorização e manutenção dos diversos ficheiros de administração.

As tarefas relacionadas com a monitorização exigem um acompanhamento quotidiano do *backbone*. A monitorização de: sistema de encaminhamento de correio electrónico, sistema de difusão das *news*, conexões UUCP, perguntas DNS, diversos servidores ("telnetd", "ftpd", etc.), espaço em disco, etc. assume um papel importante nessa administração, visto que permite detectar, diagnosticar e corrigir possíveis situações de erro. A par da monitorição existe ainda a manutenção dos ficheiros de administração, devido fundamentalmente à introdução ou alteração da configuração dos aderentes e à instalação de novos utilitários ou novas versões.

Com o objectivo de apetrechar a PTEUnet de instrumentos a monitorização foi desenvolvido um conjunto de programas. A sua utilização permite obter uma visão geral do estado do *backbone* assim como das conexões dos aderentes durante um determinado período de tempo.

Em particular, permitem elaborar estatísticas de:

- Tráfego de correio electrónico (*mail*)[1] – Apresenta uma tabela com o tráfego das mensagens de *mail* por instituição (número de mensagens e respectivo tamanho). As mensagens são contabilizadas por instituição segundo o seguinte algoritmo: se o emissor for português é contabilizada para a instituição emissora, senão caso o receptor seja português é contabilizada para a instituição receptora, caso contrário é considerado erro. Para além da separação por instituição é feita ainda a distinção entre as mensagens nacionais e internacionais, uma vez que os custos inerentes a cada uma delas são diferentes. Actualmente a cobrança aos aderentes é feita tendo em consideração unicamente a informação obtida com este comando.

- Conexões SMTP (Simple Mail Tranfer Protocol) – Apresenta uma estatística de erros nas conexões SMTP e uma outra dos diversos *mailers* utilizados, ou seja, dos protocolos utilizados para transportar as mensagens (SMTP, UUCP, etc).

- Perguntas DNS (Domain Name System)[2] – Elabora estatísticas das perguntas recebidas pelo servidor DNS a correr na "dec4pt" por: servidores que as originaram, nomes solicitados e tipo de pergunta utilizado (ver página 11).

- Conexões UUCP – Trata-se de informação bastante importante para monitorização dos aderentes que utilizam o protocolo UUCP na ligação ao *backbone*. Para cada um deles é apresentado: o número de ficheiros e respectivos bytes transmitidos em cada um dos sentidos; tempo absoluto e relativo (velocidade de transferência) utilizado na transmissão dos ficheiros; número de conecções originadas pelos aderentes e pelo *backbone* assim como o número de erros ocorridos durante as mesmas; e tempo total de "login". Por fim ainda apresenta uma estatística de ocupação das diversas *ttys* ("devices" utilizados para comunicar com o programa que implementa o protocolo UUCP – "uucico")[19, 15] assim como uma estimativa da percentagem de banda passante ocupada pelos aderentes.

- Estatística de conexões a vários servidores – Apresenta as máquinas que comunicaram (ou tentaram comunicar) com servidores como: "ftpd", "telnet", "rlogind", "nntpd", "fingerd", etc. Com isto é possível detectar tentativas de penetração no *backbone* por parte de estranhos.

Estes programas tem demonstrado ter bastante utilidade no dia a dia da administração do *backbone*, visto que permitem apresentar uma síntese da sua

[1] Implementado por Salvador Abreu (invertigador da FCT/UNL) a meados de Setembro de 91 em GNU awk e posteriormente convertido para a linguagem C por Carlos Canau (monitor do DI/FCUL).

[2] Implementado por Bryan Beecher em 89 e distribuido com o código do BIND (implementação do sistema DNS)

actividade e possibilitam uma análise rápida e bastante eficaz da mesma. Diariamente é corrido um programa que apresenta um resumo da actividade do *backbone* durante as últimas 24 horas (ver Apêndice A). Este engloba todos os programas descritos acima assim como alguns utilitários do sistema, com o objectivo de apresentar uma visão o mais abrangente possível.

Em geral, a análise de um relatório destes permite: detectar problemas no encaminhamento das mensagens de *mail*, apresentar uma visão geral sobre o tráfego de *mail* nas últimas 24 horas (tráfego distribuido por aderentes, erros ocorridos durante a entrega, etc.), apresentar uma visão geral das perguntas DNS feitas ao *backbone*, analisar o estado das conexões UUCP nas últimas 24 horas (quantidade de ficheiros e bytes transmitidos, qualidade das transmissões, etc.), analisar o espaço em disco, etc. O que é francamente melhor que a situação anterior em que se era obrigado a analisar os diferentes *logs* da véspera para verificar se havia ou não anomalias.

Muitos dos comandos descritos acima apresentam alguns problemas que advêm do facto de estarem implementados em linguagem de "shell" e "awk" e principalmente devido ao tamanho dos ficheiros de *logs* (bastante considerável). O tempo de execução de alguns deles era significativo, o que aliado à perspectiva de um contínuo aumento do tamanho dos ficheiros não apresentava um cenário muito animador. Por outro lado, o rigor de alguns dos dados obtidos não é o melhor, o que se fica a dever fundamentalmente à falta de rigor de alguma informação guardada nos *logs*. Actualmente alguns destes comandos (os mais críticos: tráfego de *mail* e conexões UUCP) estão a ser optimizados, pretendendo-se com isso melhorar os tempos de execução. No primeiro caso a conversão para a linguagem C permitiu um significativo ganho de performance (a análise diária passou a demorar menos de 2 minutos em vez dos mais de 20 minutos anteriores).

Um dos pontos mais fortes destas ferramentas é a sua grande flexibilidade o que possibilita a análise dos valores referentes a um dia, um mês ou mesmo um ano por cada um dos sub-sistemas.

4.2 Base de dados de aderentes

4.2.1 Introdução

A gestão do nó central da PTEUnet implica a parametrização de variadíssimos sub-sistemas em função dos diferentes aderentes à rede e dos diferentes interlocutores privilegiados nacionais ou estrangeiros. Essa parametrização envolve uma massa considerável de detalhes que se podem organizar por tipos de aderentes, tipos de conectividade, políticas de encaminhamento, etc. Com vista a facilitar a actividade quotidiana de gestão destes parâmetros resolveu-se organizar uma base de dados de aderentes e interlocutores. Com isso conseguiu-se automatizar a maioria dos procedimentos de parametrização do *backbone*, nomeadamente

tarefas como a introdução de novos aderentes, verificação da parametrização relativa a um determinado aderente, definição de políticas de *routing* de correio electrónico, etc.

Para que fosse possível toda esta automação foi necessário fazer uma análise detalhada do funcionamento de cada um dos sub-sistemas (correio electrónico, *news*, DNS, *routing*, UUCP, etc.) e das diversas interacções existentes entre eles, para definir quais as melhores políticas de: encaminhamento de correio, *routing* de correio, difusão das *news*, nomes UUCP, *accounting*, etc. Esta análise contribuiu para um melhor conhecimento do funcionamento de sistema e possibilitou a correcção de alguns erros cometidos até então na parametrização de alguns ficheiros de administração. Em alguns casos, constatou-se que a instalação de novos utilitários e/ou de novas versões de utilitários já instalados facilitava a definição dessas políticas. Um exemplo disso foi a instalação dos seguintes utilitários: "expect"[8, 7, 6] para permitir parametrizar o router simulando uma sessão interactiva em *batch* (visto este só poder ser parametrizado interactivamente), uma versão do "named" que regista nos *logs* todas as perguntas recebidas, Taylor UUCP por permitir utilizar o protocolo UUCP sobre TCP/IP (uma vez que o UUCP de origem do sistema operativo – Ultrix4.0 não o permitia), etc.

4.2.2 Realização

A base de dados é constituida por um conjunto de entidades, sendo cada uma das entidades definida por um conjunto de atributos (ver Apêndice B). Uma entidade pode-se definir como:

aderente – classe onde se enquadram todos os aderentes da PTEUnet.

gateway – utilizado para introduzir *gateways* de *mail*, como é o caso do "mcsun.eu.net"[3].

domínios falsos – utilizado para introduzir domínios temporários como é o caso dos: ".UUCP", ".BITNET", etc.

As entidades são definidas em ficheiros de texto, sendo estes depois convertidos para uma representação interna (ficheiros indexados – gerados através dos *library calls dbm*). A utilização desta representação visa unicamente a optimização da consulta dos dados.

Para a manipulação das diversas tabelas foram implementados vários *scrips* que permitem a geração destas, mediante os atributos definidos na base de dados. Para além disso é ainda possível verificar se as tabelas estão actualizadas e caso não estejam quais as diferenças em relação ao conteúdo da base de dados.

Assim através da base de dados é possível parametrizar os seguintes subsistemas:

[3] *Gateway* de *mail* e *news* da EUnet, situado em Amesterdão, utilizado desde há muito como ponto de referência de EUnet.

- MAIL – A definição de políticas de *routing* estático, transformação de endereços, etc. foram completamente automatizadas. A geração das tabelas, que permitem definir essas política, é feita a partir da base de dados. Este processo foi bastante simplificado devido à utilização de um servidor de encaminhamento de *mail* especial[4] que permite a separação entre as políticas e as regras de transformação de endereços, ao contrário das versões tradicionais que incluiem tudo num único ficheiro, "sendmail.cf".

- UUCP – As alterações na parametrização do UUCP relativas à introdução de novos aderentes com conectividade UUCP são feitas a partir da base de dados. Assim, a introdução de um *site* no ficheiro "sys", a criação de um *login* associado ao mesmo e a construção da árvore de directorias utilizadas como *spool* estão automatizadas. Para as executar o administrador necessita de dar apenas um comando.

- NEWS – A introdução de novos aderentes com acesso às *news* é relativamente mais complexa, uma vez que a repercussão nos ficheiros de configuração depende do modo de difusão utilizado. Este pode ser: através de NNTP (Network News Transfer Protocol) por iniciativa do aderente ou do *backbone*, por UUCP empacotando previamente os artigos em ficheiros com tamanho fixo ou através de *mail*, isto é, enviando cada artigo numa mensagem de *mail*. Em face disto, optou-se por introduzir atributos (nas entidades) que reflectissem o método pelo qual recebem as *news*.

- BIND – A introdução de novos domínios também é feita através da base de dados. Sempre que é introduzida uma nova entidade se esta tiver definidos os campos "primary" ou "secondary" os domínios respectivos serão inseridos automaticamente no ficheiro "/etc/named.boot", sem que o administrador se tenha que preocupar com isso. A geração do *master file*, para os domínios de que o servidor seja primário, é feita a partir de *templates*, uma vez que a estrutura das várias zonas dos aderentes é a mesma.

- ROUTER – Neste caso, as alterações implicam a manipulação das tabelas de *routing* definidas no *router*, "gtpuug.puug.pt". Para isso é introduzida na base de dados a informação necessária (NNA, endereço do *gateway*, endereço de rede, etc). A actualização das tabelas é feita abrindo uma sessão de "telnet" no *router* e manipulando as respectivas tabelas com comandos interactivos. Para que isso pudesse ser feito não interactivamente foi utilizado o "expect"[8, 7, 6] como foi referido atrás.

[4] Sendmail versão IDA[1].

4.2.3 Exemplo de configuração de uma entidade

A Tabela 4.1 pretende ilustrar um exemplo de configuração de uma entidade, neste caso trata-se da FCT/UNL (Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa).

Antes de começar a descrever a tabela convém referir que cada atributo é definido numa linha, sendo a primeira palavra a designação do atributo e as restantes o valor associado ao mesmo. As linhas começadas por "#" são consideradas comentários.

O primeiro atributo, `customer`, indica os serviços subscritos pela instituição, neste caso *mail* e *news*. O atributo seguinte, `remarks`, é normalmente utilizado para introduzir comentários relevantes. O nome UUCP ("unl") com que a FCT/UNL se apresenta quando estabelece conexões com o *backbone* é definido por `uucpname`. O atributo `machine` assim como os seis que se seguem são utilizados apenas como elementos informativos e constituiem no essencial a definição da FCT/UNL nos mapas UUCP[5]. Tal como os nomes sugerem designam respectivamente: hardware e sistema operativo da máquina que normalmente serve de ponto de contacto com o exterior, nome da instituição, contactos dentro da instituição, endereços electrónicos dos contactos, telefone dos contactos, morada da instituição e a sua localização (longitude/latitude). Os dois atributos seguintes, `mailer` e `mail-relay`, são utilizados na parametrização do "sendmail" do seguinte modo: todas as mensagens dirigidas ao domínio "fct.unl.pt" (ou para algum nome abaixo dele) devem ser encaminhadas por UUCP ("UUCP-A" indica que deve ser utilizado o UUCP com uma transformação de endereços especial) ao *site* "unl". O tipo de ligação é indicado pelo atributo `transport`, neste caso é utilizado UUCP sobre linha telefónica ("UUCP-DIAL-UP"). O atributo `news-transport` tem um significado semelhante mas relacionado com as *news*, "BATCH" indica que as *news* antes de serem enviadas para a FCT/UNL (por UUCP) são empacotadas em ficheiros com um tamanho fixo. Tanto estes dois atributos como os: `mail-feed` e `news-feed` são meramente informativos. No entanto, está em estudo a conjugação destes dois últimos com a política de *accounting*. Sendo actualmente utilizados para definir quais as classes de *mail* e *news* subscritas. As linhas relativas ao BIND (atributo `secondary`) indicam que a "dec4pt.puug.pt" é secundária dos domínios "fct.unl.pt", "unl.pt" "178.68.192.in-addr.arpa" e "216.68.192.in-addr.arpa". Finalmente a parte relativa à parametrização das *news* é definida pelos atributos `news-cron` e `news-sys`. O primeiro define qual a linha a introduzir no ficheiro "/etc/crontab" (aos zero minutos de todas as horas é executado o comando "sendbatches" que procederá ao empacotamento dos artigos a enviar ao *site* "unl"). O atributo `news-sys` é utilizado para definir quais as linhas que serão introduzidas no ficheiro "sys".

Supondo que a definição do aderente FCT/UNL tinha sido introduzida na base de dados, a execução do comando "make all" (na directoria corrente da base

[5] Mapas utilizados na definição de *routings* estáticos para endereços UUCP a nível mundial.

```
customer        mail
customer        news
remarks
uucpname        unl
country         pt
machine         NextCube;NeXT-Mach-2.0
organization    Fac. Ciencias e Tecnologia, Universidade Nova de Lisboa
person          Salvador Pinto Abreu
person          Vasco Pedro
email           spa@fct.unl.pt
email           vp@fct.unl.pt
telephone       +351 1 295 4464 ext. 1360
address         Quinta da Torre, PT-2825 Monte de Caparica, Portugal
location        9 08 W / 38 43 N city
mailer          UUCP-A
mail-relay      unl
mail-fee        D
# Routing
transport       UUCP-DIAL-UP
# Bind
secondary       fct.unl.pt 192.68.178.190 146.193.0.1
secondary       unl.pt 192.68.178.190 146.193.0.1
secondary       178.68.192.in-addr.arpa 192.68.178.190 146.193.0.1
secondary       216.68.192.in-addr.arpa 192.68.178.190 146.193.0.1
# News
news-fee        B
news-transport  BATCH
news-cron       0 * * * *
news-sys        unl:to.unl,\
news-sys        sci,bionet,alt,comp,news,rec,misc,soc,talk,trial,gnu,\
news-sys        eunet,pt,sci/all:f:
introduced      frazao@puug.pt  920101
```

Tabela 4.1: Definição da FCT/UNL na base de dados

de dados) levaria à modificação automática dos seguintes ficheiros de acordo com as definições da entidade FCT/UNL e as políticas definidas para o *backbone*:

- "domaintable" ("/usr/local/lib/mail/db")

```
fct.unl.pt.        unl.UUCP unl
```

  Esta informação é utilizada na conversão, pelo "sendmail", de endereços em notação UUCP para a notação DNS . Os endereço do tipo "unl!user" são transformados em "user@fct.unl.pt".

- "mailertable" ("/usr/local/lib/mail/db")

```
UUCP-A:unl          fct.unl.pt .fct.unl.pt
```

  A linha acima pretende indicar ao "sendmail" que as mensagens dirigidas a "user@<host>.fct.ul.pt" ou a "user@fct.ul.pt" devem ser encaminhadas para o *site* UUCP "unl".

- "uucpxtable" ("/usr/local/lib/mail/db")

```
unl                 fct.unl.pt
```

  A linha acima é utilizada para indicar ao "sendmail" que antes de encaminhar uma mensagem por UUCP deve converter os endereços do tipo "user@fct.unl.pt" para "unl!user".

- "aliases" ("/usr/local/lib/mail")
  O endereço "postmaster@fct.unl.pt" é acrescentado à lista de *mail* "ptunet-members" (constituida pelos utilizadores "postmaster" de todos as aderentes).

- "sys" ("/usr/local/lib/uucp")

```
system              unl
time                Never
```

  Todos os *sites* estão definidos neste ficheiro. A segunda linha indica que o *backbone* nunca toma a iniciativa de estabelecer a conexão, esta terá de vir sempre do aderente.

- "/etc/passwd"

```
Uunl::4:1:UUCP to UNL:/usr/spool/uucppublic:/usr/lib/uucp/uucico
```

  Definição da "conta" associada ao *site* UUCP "unl". A *passwd* tem de ser introduzida depois.

- "/usr/spool/uucp/sys/unl"
  Para o funcionamento do UUCP é necessário que exista uma árvore de directorias, normalmente designadas por *spool*, onde são armazenados os ficheiros que serão mais tarde transferidos (visto que o UUCP é assíncrono).

- "sys" ("/usr/local/lib/news/sys")

```
unl:to.unl,\
sci,bionet,alt,comp,news,rec,misc,soc,talk,trial,gnu,\
eunet,pt,sci/all:f:
```

  As linhas acima contêm a informação necessária para indicar quais os grupos de *news* que devem ser difundidos para a FCT/UNL.

- "/etc/crontab"

```
0 * * * * sudo -u news /usr/local/lib/newsbin/batch/sendbatches unl
```

  O comando que irá proceder ao empacotamento das *news* para as enviar por UUCP é executado a todas as horas (aos zero minutos).

- "localgroups" ("/usr/local/lib/news")

```
to.unl
```

  O grupo "to.unl" é utilizado para teste entre o *site* "unl" e o *site* "dec4pt" (*backbone*).

- "/usr/spool/news/to/unl"
  Esta directoria tem de existir, visto que é aí que os artigos destinados ao grupo "to.unl" são armazenados.

- "/usr/spool/news/out.going/togo"
  Este ficheiro é utilizado para guardar a referência dos artigos que devem ser difundidos para a FCT/UNL. É com base neste ficheiro que o comando "sendbatches" decide quais os artigos a enviar.

- "/etc/named.boot"

```
secondary  fct.unl.pt 192.68.178.190 146.193.0.1 \
           db.secondary/fct.unl.pt
secondary  unl.pt 192.68.178.190 146.193.0.1 \
           db.secondary/unl.pt
secondary  178.68.192.in-addr.arpa 192.68.178.190 146.193.0.1 \
           db.secondary/192.68.178
secondary  216.68.192.in-addr.arpa 192.68.178.190 146.193.0.1 \
           db.secondary/192.68.216
```

As linhas acima indicam que o servidor BIND da "dec4pt" é secundário das zonas: "fct.unl.pt", "216.68.192.in-addr.arpa", "178.68.192.in-addr.arpa" e "unl.pt".

Para além destes ficheiros, a base de dados pode ainda manipular os seguintes ficheiros:

- "nntp_access" ("/usr/local/lib/news")
  É utilizado sempre que se pretenda difundir as *news* por NNTP
- *master file* ("/var/named/db")
  Definição das zonas de que o servidor BIND da "dec4pt" é primário.
- "gt-lisbon-confg" ("/home/admin/cisco")
  Este ficheiro contém a parametrização do *router* que aceita as conexões dos aderentes que pretendem ligar ao *backbone* por TCP/IP.

Assim qualquer novo aderente, ou a modificação de um atributo de um aderente já existente pode ser repercutida nos ficheiros de administração do *backbone* de forma completamente automática.

4.2.4 Conclusão

A prática tem demonstrado que a utilização da base de dados apresenta bastantes vantajens. De facto, para além da rapidez com que se passou a fazer a introdução de um novo aderente, problemas como o esquecimento da alteração de uma ou outra tabela, protecções não válidas em alguns ficheiros, etc. deixaram de ser preocupantes. Com isso conseguiu-se melhorar o serviço prestado aos aderentes mas principalmente simplificar a administração do *backbone*.

Uma futura restruturação do sistema de *accounting* passará em princípio pela sua interligação com a base de dados.

4.3 Name system

4.3.1 Introdução

O *name system* é uma espécie de base de dados distribuída capaz de associar nomes a recursos. No Internet o *name system* mais conhecido e utilizado é o DNS (ver Capítulo 3). Uma das características que mais contribuiu para o seu sucesso foi a sua gestão distribuída e descentralizada. De facto esta permite uma administração local dos recursos e portanto bastante próxima da sua localização física e/ou administrativa o que facilita a sua gestão. Esta descentralização da gestão exige uma delegação de autoridade para partes da árvore de nomes (ver página 19) o que leva a que muitos dos administradores nem sempre tenham experiência suficiente para a função que desempenham conduzindo-os muitas vezes à introdução de inconsistências na base de dados.

De facto, um administrador de uma zona DNS para além dos inevitáveis conhecimentos teóricos sobre o assunto deve ter uma grande componente prática, sob pena de cometer erros com consequências mais ou menos graves para o correcto funcionamento do sistema.

A minha experiência nesta área aponta exactamente para isso. Inicialmente quando comecei a acompanhar a administração das zonas "ul.pt", "puug.pt" e de alguns aderentes da PTEUnet, ainda não tinha sensibilidade para alguns aspectos que à primeira vista não parecem importantes mas que na prática assumem um papel fundamental no correcto funcionamento do sistema. Muitas vezes só através do contacto com situações verídicas ou através do diálogo com administradores mais experientes se consegue ganhar essa sensibilidade.

Um outro ponto importante é a detecção e correcção de inconsistências. Muitas vezes estas só são detectadas quando os problemas que provocam são de tal maneira graves que levam ao não funcionamento de alguma funcionalidade importante.

4.3.2 Montagem de .PT

A montagem do DNS português é relativamente recente (meados de 91). Isto deve-se fundamentalmente à inexistência de uma infra-estrutura IP em condições até então. Contudo com o surgimento do projecto "Serciço IP da RCCN"[10] essa infra-estrutura IP começou a ser uma realidade o que levou à consequente montagem do DNS nacional.

Numa primeira fase, começou-se por montar servidores na maioria das universidades participantes no projecto e no INESC, criando assim um sistema autónomo (com uma "."–root falsa). Esta fase foi bastante propícia à obtenção da experiência referida atrás. Para isso muito contribuiram os diversos erros cometidos e posteriormente corrigidos. Dos quais se destaca um que, pelas suas consequências, assumiu um papel importante na assimilação de certos cuidados a ter na gestão de uma sistema deste tipo. Na sua origem esteve a alteração de endereços no INESC/Lisboa que aliado a uma incorrecta definição de parâmetros e com uma certa falta de coordenação gerou um ciclo entre 3 *routers* (INESC, PUUG e UNL) levando inclusive à saturação das linhas.

Com a ligação IP para o exterior consumada no Outono de 1991 passou-se a uma fase em que os servidores nacionais podiam interrogar os estrangeiros e vice-versa. Nesta fase, o nosso sistema deixou de ser autónomo para se ligar, ainda que pacialmente, ao sistema mundial. Para isso os diversos servidores portuguêses passaram a reconhecer os verdadeiros servidores de "." (root). Por outro lado, para que fosse possível aos servidores estrangeiros obter respostas de nomes nacionais seria necessário alterar a lista de servidores registados no NIC (Network Information Center). Enquanto isto não foi possível (só no inicio de 92 a lista de servidores anunciados pela zona "." passaram a ser os actuais), encontrou-se uma solução temporária, primeiro forçando um dos servidores oficiais da zona .PT, "mcsun.eu.net", a importar a nova zona e mais tarde levando

os restantes a tomar a mesma atitude. De facto, muito antes do NIC actualizar a lista de servidores já o DNS português estava totalmente integrado no mundial. Para isso contamos com o precioso auxílio da EUnet, que para além de começarem por importar a zona .PT também nos serviram de cartão de apresentação no exterior, nomeadamente na obtenção de servidores para .PT fora do país. De igual modo, também os conselhos e a análise feitos por Piet Biertema e Daniel Karrenberg (dos serviços centrais da EUnet) permitiram corrigir muitos dos erros que iam sendo cometidos, fruto da nossa falta de experiência.

Durante este processo de montagem do DNS, foi criado um grupo de trabalho, "fccn-ip-wg" (FCCN IP Working Group) com o objectivo de permitir entre outras coisa a partilha de experiências e a coordenação de esforços para a montagem do IP em Portugal. A minha colaboração com o PUUG e o DI/FCUL levaram a que integrasse este grupo (desde meados de 91), no qual assumo as funções de co-administrador da zona .PT em paralelo com Prof. Legatheaux Martins e o Dr. António Inês do LNEC.

Actualmente o DNS português é já uma realidade, verificando-se um constante crescimento da quantidade de recursos por ele definidos.

4.3.3 DDT

Uma das questões mais delicadas do DNS é sua dificuldade de detecção de inconsistências. Na maioria dos casos estas só são detectadas quando inviabilizam a obtenção dos fins para que a informação em causa foi introduzida. Isto leva a que muitas delas perdurem durante algum tempo sem que sejam detectadas, com todos os inconvenientes daí resultantes. Actualmente cabe ao administrador a tarefa de analisar a(s) zona(s) que administra. Contudo o problema pode ser ainda mais complexo, uma vez que é frequente existir informação cruzada (relativa a outras zonas), o que obrigaria a uma maior coordenação de esforços entre os administradores envolvidos.

A percepção desta situação aliada à falta de ferramentas que auxiliem o administrador neste tipo de funções levou ao desenvolvimento de um projecto que tinha como principal objectivo prestar esse auxílio.

O projecto foi denominado Ddt (Domain Debug Tools), sob o qual se pretendia implementar um conjunto de ferramentas que possibilitassem uma análise sistemática de uma parte da árvore DNS (ver Capítulo 5). O seu desenvolvimento conheceu quatro fases distintas. Numa primeira fase foi feita a análise e implementação dos comandos. A análise foi baseada fundamentalmente na minha experiência e em alguma documentação informal existente. Ainda nesta fase foram feitos alguns testes sobre o DNS português, sendo as conclusão, relativamente à sua utilidade, bastante favoráveis. A fase seguinte foi dedicada à redacção dos manuais, feita em inglês. Com isto pretendia-se que a utilização deste pacote não estivesse limitada por barreiras liguísticas. Numa terceira fase foi feita uma análise exaustiva do DNS europeu utilizando o Ddt. Os resultados foram bastante animadores, principalmente quando se trata de análise de peque-

nas partes da árvore. Para a análise de países como a Alemanha verificaram-se alguns problemas directamente relacionados com quantidade de informação a tratar. Por fim redigi um artigo em conjunto com o Prof. Legatheaux onde para além da apresentação do Ddt é feita uma introdução ao DNS e são apresentadas as conclusões da análise do DNS europeu.

Durante toda a execução do projecto foi importante o *feedback* estabelecido com o orientador do estágio (Prof. Legatheaux Martins), tanto durante a análise como na elaboração da documentação.

A expectativa quanto à aceitação deste pacote é enorme. Neste momento encontra-se disponível nos *archives*: "mcsun.eu.net" (EUnet), "ftp.nluug.nl" (NLnet) e "ftp.puug.pt" (PTEUnet).

Por outro lado, o artigo referido atrás foi recentemente proposto para a conferência NSC'92[6].

4.4 Archive

A montagem de um *archive server* (arquivo de ficheiros) na PTEUnet constava na proposta inicial do estágio, contudo foi adiado para depois da defesa por falta de tempo para a sua realização. Apesar disso foi feita uma análise preliminar na qual se pretendeu estudar qual a estrutura mais adequada para o mesmo e foi feito um levantamento daquilo que tem sido feito nesta área.

O *archive* deverá obedecer aos seguintes requisitos: os custos de manutenção devem ser o mais reduzidos possível, deve disponibilizar *interface* para FTP, *mail* e possivelmente para UUCP e deve permitir definir uma política de protecções e *accounting*.

Para simplificar o estudo, considerou-se um sistema deste tipo como sendo constituido pelas seguintes componentes:

1. Actualização e manutenção – Num sistema deste tipo manter os ficheiros actualizados requer elevados custos. Na maioria dos *archives* existentes é ao administrador que cabe esta tarefa, isto é, este obtem informação (de alguma maneira) da existência de novas versões ou de novos utilitários e copia os ficheiros respectivos, normalmente por FTP, para o arquivo local. No caso de se tratar de um *archive* relativo um tema específico esta tarefa pode não ser muito significativa, podendo ser feita em *part-time*. No entanto, se não tiver restrições deste tipo, torna-se um tarefa demasiado dispendiosa. Por isso a sua automatização (ou semi-automatização) pode assumir um papel importante para o seu sucesso a médio e longo prazo.

2. Classificação e pesquisa – Estes dois aspectos estão intimamente relacionados com o método utilizado para a organização dos ficheiros. Neste caso,

[6] NSC'92 – *The Network Services Conference 1992*, Pisa, Itália, 3 a 5 de Novembro de 1992.

a classificação pode depender do método utilizado na actualização do arquivo.

3. *Interface* – Um sistema deste tipo pode disponibilizar vários tipos de *interfaces*. Os mais comuns são os *interfaces* para: FTP, *mail* e UUCP.

4. Protecção e *accounting* – A integração de um sistema deste tipo na PTE-Unet, passaria por distribuir os seus custos pelos aderentes. Para isso seria necessário implementar um mecanismo de protecções e *accounting* para permitir contabilizar a sua utilização.

A primeira abordagem realizada permitiu constatar que o desenvolvimento de um sistema deste tipo (que obedeça minimamente aos requisitos propostos) requer, fundamentalmente, um grande investimento no aperfeiçoamento e automatização da primeira componente (actualização e manutenção), visto ser esta a que consome a maior parte dos custos de administração.

Da investigação dos sistemas existentes, foram encontrados vários utilitários que poderão vir a ser utilizados, dos quais se destacam os seguintes:

- "mserv" – servidor que permite responder a pedidos de ficheiros por *mail*.
- "rkive" – comando que permite arquivar os artigos das *news*.
- "prospero"[14] – *file system* distribuido sobre qual deverá ser implementado o *archive server* da EUnet.

Uma possível automatização da actualização e manutenção dos ficheiros poderá passar por uma interligação com o sistema "archie"[5][7], uma vez que este dispõe de informação sobre quais os utilitários existentes e que *archives* os disponibilizam.

Por outro lado, a própria EUnet formou um grupo de trabalho com o objectivo de encontrar a melhor solução para montagem um *archive* da EUnet que, de alguma forma, englobasse os diversos *archives* já existentes nos ramos nacionais. Em face disso, o diálogo com esse grupo poderá ser importante, pois permitirá partilhar a experiência obtida.

[7] Sistema que contém informação relativa a diversos *archives* distribuidos por todo Internet mundial, nomeadamente os ficheiros que estes disponibilizam.

Capítulo 5

Domain Debug Tools

5.1 Introdução

Na administração do DNS é muito frequente a introdução de erros, quer pela inexperiência de alguns administradores (consequência directa das sucessivas delegações feitas ao longo da árvore), quer por esquecimento ou falta de cuidado. Podemos enumerar alguns dos mais frequentes:

- Definição dos parâmetros do SOA RR (*timers*) não ajustaveis à situação. Se forem demasiado elevedos a frequência dos *updates* é menor o que leva a que as inconsistências perdurem mais tempo no sistema. Por outro lado, os valores demasiado baixos podem provocar uma sobrecarga nas linhas de comunicação (principalmente se estas forem de baixa qualidade).
- Incorrecta actualização do número de versão ou o esquecimento de o actualizar aquando da alteração da zona.
- Introdução de *glue record* desnecessários.
- Definição de políticas de *routing* ilegais, ou porque introduzem problemas no encaminhamento do *mail*, ou então porque utilizam máquinas como *gateways* sem a prévia autorização dos administradores destas.
- Diferenças entre os servidores indicados como sendo autoritários e os que na realidade o são.
- Utilização de RRs com um semântica diferente da que está definida.
- Referências para nomes inexistentes, como são os casos de aliases ou MX RRs que apontam para *hosts* inválidos.
- O célebre *missing trailing dot problem*, ou seja, o esquecimento do ponto à direita dos nomes aquando da sua definição no *master file*.

Muitas destas inconsistências têm um tempo de vida relativamente longo, principalmente por serem difíceis de detectar (pelo menos à primeira vista) e de não existirem ferramentas que auxiliem nessa tarefa. Em face disso desenvolvi um projecto denominado *Ddt (Domain Debug Tools)* que teve como objectivo principal a realização de um conjunto de ferramentas que permitesse verificar a coerência de uma ou mais zonas. A sua utilização permite aos administradores detectar e corrigir muitos dos mais frequentes (e mais graves) erros habitualmente introduzidos no DNS. Para além disso, permite ainda a obtenção de uma vasta gama de rácios sobre o DNS, tais como: nomes mais populares, *hosts* por domínio, etc.

O Ddt é constituido por um conjunto de comando que correm sobre *master files* armazenados localmente. À primeira vista pode parecer que a opção de trabalhar sobre a cópia dos dados não garante a sua actualização aquando da análise. De facto isso é verdade, contudo com a interrogação *on-line* dos servidores o problema seria ainda mais grave, visto ser impossível assegurar que os dados não são alterados durante uma análise (entre duas *queries*) o que pode levar à detecção de inconsistências que na realidade não existem. Por outro lado, dada a própria filosofia da gestão da informação as consequências causadas pela possível desactualização dos dados são perfeitamente suportáveis. Como consequência de se trabalhar sobre dados locais a análise é muito mais eficiente, sendo mesmo a única opção aceitável quando as comunicações são de má qualidade.

Para construir o *cache* das diversas zonas pode ser utilizado o comando "ddt-xfer"[1]. Este comando permite transferir uma zona e eventualmente todas as suas subzonas. Pode-se inclusivamente transferir toda a árvore de DNS com um único comando, isto obviamente se se possuir espaço em disco e banda passante suficientes (por outro lado, o tempo necessário será tal que depois de terminar, a maior parte da informação já estará desactualizada). À semelhança do que é feito pelo "named-xfer" as zonas só são transferidas se e só se a versão armazenada localmente estiver desactualizada (número de versão inferior).

Os restantes comandos analizam a definição de zona (SOA e NS RRs), *glue records*, MX RRs, *reverse mapping*, RRs em geral, etc. Cada um destes comandos pode analizar uma zona ou um conjunto de zonas em simultâneo, como um país, um continente, etc.

Com o objectivo de facilitar a interpretação das mensagens de erro, estas foram classificadas em quatro níveis consoante a sua gravidade. Assim temos os seguintes níveis:

1. [Warning 1] – Deve investigar isto já!
2. [Warning 2] – Não se esqueça deste aviso!
3. [Warning 3] – Tem de analisar isto algum dia!

[1] Versão ligeiramente modificada do "named-xfer" (comando do BIND) para facilitar a obtenção de partes da árvore DNS.

4. [Comment] – O Ddt não tem informação suficiente para tomar a decisão exacta.

As mensagens incluidas no nível [Warning 1] são as mais graves. Por outro lado, o nível [Comment] é o menos importante. No entanto não quer isto dizer que devam ser ignoradas. Muitos dos erros são aí incluidos porque o Ddt não dispõe de informação suficiente para tomar uma decisão exacta.

No próximo capítulo são apresentados os comandos disponíveis no pacote Ddt.

5.2 Comandos disponíveis

ddt-xfer

O comando "ddt-xfer" permite transferir as zonas para o *cache* local para que sejam mais tarde analizadas pelos restantes comandos. No exemplo abaixo pretende-se exemplificar a transferência da zona "puug.pt" (caso a de versão seja diferente da armazenada no *cache*) de um dos servidores "dec4pt.puug.pt." ou "inesc.inesc.pt.". Esta será guardada num ficheiro com o mesmo nome na directoria corrente.

```
ddt-xfer -z puug.pt -f . dec4pt.puug.pt. inesc.inesc.pt.
```

Se se pretender transferir também todas as suas subzonas, o parametro "-r" terá de ser incluído:

```
ddt-xfer -z puug.pt -f . -r dec4pt.puug.pt. inesc.inesc.pt.
```

soac

O comando "soac" (Start Of Authority Check) permite analizar o SOA RR que define o início da zona e a lista dos seus NS RRs. Relativamente ao SOA RR verifica se os *timers* (página 12) estão dentro de um determinado intervalo. Para permitir uma certa flexibilidade tanto os valores correctos como os intervalos aceitaveis são definidos numa tabela à parte, ".SOA-timers". Para além disso verifica ainda se os servidores indicados na definição de zona (lista de NS RRs) são na verdade autoritários sobre a mesma. Em simultâneo é possível saber se a versão corrida por cada um dos servidores corresponde à actualmente no *cache* assim como se algum deles estiver inacessível isso também será apresentado. No exemplo apresentado o parâmetro REFRESH é bastante baixo (6 segundos). Estas situações tornam-se mais sérias quando associadas à utilização de linhas de baixa qualidade. A segunda mensagem pode-se considerar mais grave, uma vez que o *host* "colombo.puug.pt" é apresentado como sendo responsável pelo domínio "puug.pt" o que na realidade não acontece.

```
$ soac puug.pt
SOA Record
Refresh     360    [recommend value: 28800]
Authoritative servers
colombo.puug.pt   [Not authoritative answer]
```

rrc

O comando "rrc" (Resource Records Check) permite detectar muitos dos erros, sintáticos e semânticos, introduzidos aquando da definição dos RRs. Assim erros como: referências a nomes inválidos, esquecimento do ponto à direita dos nomes aquando da sua definição no *master file* (conhecido como *missing trailing dot*), utilização incorrecta do nome "localhost", etc. são apanhados. Todos os erros referentes a nomes externos à zona são incluídos no nível [Comment].

A primeira mensagem no exemplo apresentado é provocada por um *missing trailing dot* no nome "gtpuug.puug.pt". A segunda mensagem indica que o *host* deveria ter sido definido em notação DNS e não com o endereço IP. Por fim, a duas últimas indicam que existe referência para os *hosts* "bach.puug.pt" e "inesc.inesc.pt" na zona "puug.pt" mas que não foram encontrados os seus endereço IP (A RR). A primeira delas é mais delicada, visto que se trata de um nome pertencente à zona e portanto deveria de ter sido encontrado. Pelo contrário, a segunda pode ter sido provocada apenas por falta de informação.

```
$ rrc puug.pt
Perharps name without trailing dot: gtpuug.pt.puug.pt.puut.pt
Hostname 192.67.76.4 should be in domain notation
No A/CNAME record found for bach.puug.pt
No A/CNAME record found for inesc.inesc.pt [Can't verify it]
```

grc

Os *Glue records*[2] são muitas vezes responsáveis pela introdução de inconsistência no DNS. Para detectar a sua presença foi implementado o comando "grc" (Glue Record Check). O facto de ser detectado um *glue record* numa zona não significa necessariamente que este esteja definido no verdadeiro *master file*. Na realidade para que este tipo de *records* seja transferido junto com os restantes RRs da zona é suficiente que o servidor de quem foi feita a transferência seja autoritário sobre eles.

Os *glue record* podem ser de dois tipos: necessários ou desnecessários. No primeiro caso estão incluidos aqueles que definem o endereço para um servidor de uma subzona estando eles próprios dentro de uma subzona. A sua introdução é obrigatória, visto serem necessários para descer na árvore (página 20). Caso contrário, não devem ser introduzidos junto com a restante informação no *master*

[2] *Glue record* é um A RR associado a um servidor que não pertence à zona onde lhe é feita a referência.

file. No exemplo abaixo a primeira mensagem indica a preseça de um *glue record* necessário, visto que assume um papel primordial na ligação ("colagem") das zonas: "unl.pt" e "fct.unl.pt". Por outro lado, o segundo *glue record* encontrado não é necessário, uma vez que está acima da zona "unl.pt"

```
$ grc unl.pt
Necessary glue record. Name server of delegated domains.
trinstan.fct.unl.pt     IN  A    192.68.178.190
Unnecessary glue record. If present, should be removed.
inesc.inesc.pt          IN  A    146.193.0.1
```

mxc

Com o objectivo de evidenciar políticas de *mail routing* estranhas ou pelo menos menos usuais foi desenvolvido o comando "mxc" (Mail eXchange Check). Dado ser demasiado complexo definir qual a política de *mail* ideal para determinada instituição/país optou-se por apresentar apenas as políticas que à primeira vista se apresentariam como sendo menos normais. Ou seja, sempre que há MX RRs a apontar para fora da zona. Dentro destes é dado maior ênfase às que definem MX RRs a apontar para *hosts* dentros da zona com preferência superior a outros que apontam para fora. É o caso do exemplo abaixo. Por além do papel desempenhado na detecção de inconsistências é ainda possível utilizá-lo para evidênciar a(s) política(s) de *mail routing* utilizadas numa região, país, etc.

```
$mxc puug.pt
MX record pointing to an external host.
puug.pt         IN  MX  20  inesc.inesc.pt
MX record to local or delegated host after MX record to external host.
puug.pt         IN  MX  30  colombo.puug.pt
```

rmc

Finalmente, o comando "rmc" (Reverse Mapping Check) verifica a coerência dos *reverse mappings*. A cada PTR RR deve estar associado um A RR e vice-versa. Com este comando é possível verificar se existem incoerências entre estes dois tipos de informação. No exemplo seguinte são apresentadas as duas situações. No primeiro caso um PTR RR não encontrado, em princípio porque a zona "76.67.192.in-addr.arpa" não foi analizada. E por último a referência a um *host* "tspuug.puug.pt" com o endereço "192.84.62.254" sem que exista um A RR associado ao mesmo.

```
$ rmc puug.pt 62.84.192.in-addr.arpa
## PUUG.PT ##
No 4.76.67.192.in-addr.arpa PTR RR found for colombo.puug.pt.

## 62.84.192.IN-ADDR.ARPA ##
No tspuug.puug.pt. A RR found with 192.84.62.254
```

5.3 Rácios

Para além dos comandos de análise de DNS foram também desenvolvidos uma séria de *scripts* que permitem obter vários rácios sobre a informação armazenada no DNS. Através da sua manipulação é possível obter: o número de *hosts* por domínio ou por rede, os nomes mais populares, os *gateways* de *mail* mais populares, etc. Os diversos *scripts* foram realizados de modo a oferecerem uma grande flexibilidade para a obtenção das mais variadíssimas estatísticas. Por outro lado, dado a sua simplicidade é relativamente fácil introduzir outros *scripts* sempre que se pretenda um filtro, uma formatação, etc. não prevista.

O comando seguinte apresenta a lista de nomes de máquinas em Portugal ordenadas pela sua popularidade:

```
expand /usr/local/ddt.cache/pt/* | host | names-stat | sort -rn
```

O *script* "expand" normaliza todos os nomes, substituindo todos os nomes relativos por nomes absolutos. O *script* "host" filtra os nomes das máquinas a partir dos A RRs e finalmente o *script* "names-stat" elabora um rácio de ocorrências dos diversos nomes.

O próximo comando elabora uma tabela com o número de *hosts* por rede em Itália.

```
expand /usr/local/ddt.cache/it/* | hosts-addr | nets | sort -rn
```

O *script* "hosts-addr" filtra os nomes das máquinas com o respectivo endereço IP e o "nets" elabora a tabela com o número de *hosts* por cada uma das redes.

5.4 Análise do DNS europeu

Durante a semana de 13 a 20 de Abril foi feita uma análise exaustiva ao DNS europeu. Desta análise foram obtidos várias conclusões sobre os erros mais frequentes na Europa assim como resultados interessantes sobre os vários "estilos" de administração do DNS em diferentes países e regiões. Neste capítulo vão ser apresentados muitos destes resultados, alguns deles sob a forma de tabelas.

Os factos apresentados baseiam-se no estado do DNS na noite de 13 para 14 de Abril, ou seja, na data em que foi construído o *cache* local dos vários ramos europeus da ávore DNS. Muitos dos erros encontrados já devem de ter sido corrigidos assim como também devem ter sido cometidos muito mais.

Da análise podemos constatar que, de um modo geral, os administradores do DNS seguem as opções tomadas pelo administrador do domínio superior do país (*top-level domain*) ou dos domínios mais conhecidos.. Isto conduz a uma uniformização dos hábitos dentro de uma região e mesmo dentro de um país. Um exemplo disso são as estratégias de construção de um número de série sempre crescente (Ver Tabela 5.3).

A definição dos *timers* do SOA apresenta uma distribuição que não deixa de ser curiosa. Em média têm valores mais elevados nos países do sul. Isto deve ser o reflexo do preço e qualidade das linhas que em geral esses países têm, em comparação com os do centro e norte da Europa.

Um outro aspecto engraçado são as diferenças políticas de designação nos diferentes países. A Alemanha destaca-se dos restantes pelo comprimento dos seus nomes. De facto é a campeã absoluta dos nomes mais compridos. Nada mais nada menos que os duzentos maiores nomes de máquinas na Europa estão debaixo do domínio ".de". Por outro lado, o nome de máquina mais curto é italiano ("it" com o endereço 131.114.1.30).

A Tabela 5.1 apresenta uma série de rácios relativos a cada um dos países analisados. As três primeiras colunas apresentam respectivamente: o número de zonas, número de domínios e número de *hosts* por país. A quarta coluna apresenta o espaço em disco ocupado pelas zonas de cada um deles. A quinta, sexta e sétima combinam as três primeiras, apresentando: domínios por zona, *hosts* por zona e média de ocupação em disco por zona. Por fim as duas últimas colunas indicam a profundidade média e máxima da definição das zonas.

	Zonas	Doms	Hosts	Kbytes	Doms Zona	Hosts Zona	Kbytes Zona	Comp-Zona	
								Média	Max
AT	99	150	4345	459	1.5	44	4.6	3.65	5
BE	15	187	1043	133	12.5	70	8.9	3.53	4
BG	1	7	0	3	7.0	0	3.0	1.00	1
CH	37	218	13367	1143	5.9	361	30.9	2.00	3
CS	4	32	177	15	8.0	44	3.8	2.00	3
DE	651	2316	38376	4737	3.6	59	7.3	3.00	5
DK	64	241	2319	353	3.8	36	5.5	2.38	4
ES	120	393	3086	330	3.3	26	2.8	2.05	3
FI	108	346	14493	1147	3.2	134	10.6	2.56	4
FR	254	427	16024	1643	1.7	63	6.5	2.48	4
GR	41	49	541	109	1.2	13	2.7	2.51	3
HU	2	8	5	4	4.0	2	2.0	2.00	3
IE	16	64	515	87	4.0	32	5.4	2.31	3
IL	26	42	2552	148	1.6	98	5.7	3.31	4
IS	6	10	285	22	1.7	48	3.7	2.33	3
IT	127	317	3947	1627	2.5	31	12.8	2.88	4
NL	230	529	14583	1578	2.3	63	6.9	2.94	4
NO	243	357	12808	1037	1.5	53	4.3	2.65	4
PL	31	37	324	74	1.2	10	2.4	2.94	4
PT	43	48	1158	114	1.1	27	2.7	2.31	3
SE	262	1803	18836	2200	6.9	72	8.4	2.79	6
SU	1	105	0	12	105.0	0	12.0	1.00	1
TN	2	3	8	6	1.5	4	3.0	1.50	2
UK	191	1027	24371	2463	5.4	128	12.9	3.60	5
YU	2	18	13	7	9.0	6	3.5	1.50	2

Tabela 5.1: Diversos rácios por país (14/4/1992)

5.4.1 Estrutura da árvore

Apenas quatro dos países europeus optaram por uma estrutura de árvore com domínios "AC", "CO", "GOV", etc. imediatamente abaixo do domínio superior do país, definindo as organizações apenas num terceiro nível. São eles: Austria, Bélgica, Israel e Reino Unido. Todos os outros optaram por definir as organizações a partir do segundo nível.

Um outro resultado interessante é a profundidade na árvore a que são feitas delegações de autoridade. Os casos extremos são a Bulgaria e CIE (antiga União Soviética) com um nível apenas (existe uma única zona) e a Suécia com três zonas definidas ao sexto nível ("adm.hj.se.adm.hj.se", "tekn.hj.se.tekn.hj.se" e "uki.hj.se.uki.hj.se" – tudo indica que o seu aparecimento se ficou a dever ao "célebre" *missing trailing dot*).

5.4.2 Timers do SOA

Tal como já foi referido atrás os países do sul utilizam *timers* de um modo geral mais elevados. Na Tabela 5.2 podem ser vistos os timers médios utilizados em cada um dos países.

Mais interessante, mas também mais grave, são os valores utilizados em algumas zonas:

- Valor do EXPIRE superior a um ano (36000000 segundos).
- Os quatro valores iguais a 15 segundos ("silab.dsi.uniti.it").
- Valor do DEFAULT TTL nulo ("agw.docs.uu.se").

5.4.3 Números de série

A Tabela 5.3 apresenta as políticas mais utilizados em cada um dos países de modo a assegurar que um número de série sempre crescente. As estratégias mais utilizadas são: simplemente incremental (começando em 1, sendo incrementado de uma unidade), decimal (Número.Versão – por exemplo 1.01) e codificação da data (com várias variantes: 2 ou 4 dígitos para o ano (à esquerda), utilizando zero ou mais dígitos (à direita) para distinguir a versão no mesmo dia, etc).

Há no entanto muitos administradores que aparentemente não seguem nenhuma heurística, como são os casos de "iiasa.ac.at" com 2147483647, "epfl.ch" com 587000483, etc.

Um erro cometido frequentemente é a diminuição do número de série. Durante a análise podemos verificar que era provocado, na maioria das situações, por:

- Alteração da estratégia de construção do número de série sem tomar as devidas precauções

	Refresh	Retry	Expire	Default TTL
AT	21058	3074	708541	69927
BE	6720	1480	2528640	195840
BG	86400	14400	2592000	345600
CH	21873	4092	1479989	10657
CS	67500	26100	10656000	280800
DE	22227	2663	2427647	95146
DK	11095	3976	3526706	77634
ES	28050	6840	677040	88200
FI	18300	1569	1983267	44444
FR	22632	4886	3381676	166468
GR	38371	13580	2937600	198088
HU	86400	9000	3096000	216000
IE	41625	6825	1150200	153787
IL	30750	7502	2666215	115477
IS	28800	7200	604800	86400
IT	42553	2489	1698208	132718
NL	20413	6098	1028871	84346
NO	11956	1260	1347906	76274
PL	69213	6155	2508387	195097
PT	72686	7200	1357714	140914
SE	10538	1773	2668534	69703
SU	86400	14400	2592000	345600
TN	50400	9000	734400	129600
UK	12187	3777	1725540	80915
YU	28800	7200	604800	86400

Tabela 5.2: *Timers* SOA médios por país (14/4/1992)

- Utilização da data de forma incorrecta. Foram encontrados pelo menos duas formas de codificar a data de maneira que não garante sempre o crescimento: codificação por uma ordem incorrecta (50392 – "ariadne-t.gr", 150192 – "hitec.ariadne-t.gr", 40292 – "nrcps.ariadne-t.gr", todos estes números de série parecem ter sido construídos utilizando a data da seguinte forma DDMMYY); a outra situação é a utilização de um único dígito para codificar o dia ou o mês quando estes são inferiores a 10 (199112501 – "c3consult.comm.se").

- Actualização defeituosa do número em notação decimal. Por exemplo, se número 1.31 for alterado para 1.4 sofrerá uma regressão, visto que o ponto será substituido por três zeros e portanto o número 100031 passará para 10004 (que é obviamente inferior).

Para além deste tipo de distracções existe ainda uma outra que também pode trazer problemas para o crescimento do número de série, é o caso do

número negativo. Foram encontrados vários servidores com versões de zonas com o número de série negativo. Na maioria das situações tratava-se apenas de um dos servidores autoritários, o que leva a pensar que as inconsistências já subsistem há algum tempo.

5.4.4 Servidores

Como seria de prever, foram encontradas várias discrepâncias entre a lista de servidores presente na zona e os servidores que são realmente autoritários sobre a mesma. Um dos casos mais curiosos é a zona "tmt.tele.fi" que apresenta "." (sim é mesmo "root"!) como um dos seus servidores.

Na Tabela 5.4 são apresentados os 20 servidores europeus mais populares.

5.4.5 Utilização de MX RRs

Os MX RRs são realmente muito úteis. Encontramos 130994 em toda a base de dados europeia. Obviamente, isto representa centenas de políticas de *mail routing*. No entanto, alguns deles (algumas centenas) apontam para máquinas desconhecidas! Um aspecto curiosos entre as diversas políticas é o facto da França ser o único país com nenhum MX RR a apontar para uma máquina fora do país. Isto não significa que todos os MX RRs estejam bem definidos. De facto, pelo menos 437 apontam para máquinas incorrectas (aparentemente devido ao *missing trailing dot*).

Finalmente, é ainda interessante conhecer os 20 *gateways* de *mail* mais populares na Europa (ver Tabela 5.5).

5.4.6 "The missing trailing dot"

Este é um dos problemas mais sérios, principalmente pela frequencia com que aparece. Neste campo os franceses parecem ser os campeões com nada mais nada menos que 532 de nomes com fortes suspeitas de serem provocados por este problema. No entanto, o campeão absoluto (destacadíssimo) é a zona "ircam.fr" com 428, 214 em MX RRs apontando para "vaslav.ircam.fr.ircam.fr" e os outros 214 em MX RRs apontando para "nadia.ircam.fr.ircam.fr".

Uma interessante repercussão deste erro é o aparecimento das zonas "adm.hj.se.adm.hj.se", "tekn.hj.se.tekn.hj.se" e "uki.hj.se.uki.hj.se" como já foi referido atrás.

5.4.7 Outros aspectos curiosos

É possível encontrar vários nomes engraçados no DNS, tanto devido a negligência como ao sentido de humor de alguns administradores. Um destes casos é o MX RR de "bibsys.unit.no" que aponta para "taken.out.of.service.unit.no". De igual modo, tudo indica que o nome "#gnasher.cineca.it" tenha uma origem

	Simplesmente incremental	Décimal (com ponto)	Codificação da data
AT	muitas	poucas	poucas
BE	nenhuma	poucas	muitas[a]
BG	nenhuma	todas	nenhuma
CH	muitas	algumas	poucas
CS	algumas	algumas	algumas
DE	algumas	algumas	muitas[b]
DK	poucas	algumas	muitas
ES	poucas	nenhuma	muitas[c]
FI	poucas	poucas	muitas[d]
FR	poucas	poucas	muitas[e]
GR	algumas	muitas	nenhuma
HU	nenhuma	algumas	algumas
IE	algumas	nenhuma	algumas
IL	algumas	poucas	nenhuma
IS	nenhuma	nenhuma	todas[f]
IT	algumas	poucas	muitas[g]
NL	algumas	algumas	muitas
NO	poucas	muitas[h]	poucas
PL	poucas	poucas	muitas[i]
PT	algumas	nenhuma	muitas
SE	poucas	poucas	muitas[j]
SU	nenhuma	todas	nenhuma
TN	nenhuma	todas	nenhuma
UK	poucas	poucas	muitas[k]
YU	todas	nenhuma	nenhuma

Tabela 5.3: Políticas de construção do número de versão (14/4/1992)

[a] YYMMDDV – mais utilizado.

[b] São utilizados vário formatos, é no entanto de destacar a utilização por algumas zonas do número 53 (porto do serviço *domain*) à esquerda da codificação da data).

[c] YYMMDDVVV – utilizado por quase todas as zonas .

[d] YYYYMMDDVV – mais utilizado.

[e] YYMMDDVV E YYMMDDVS – utilizados com maior frequência.

[f] YYYYMMDDVV – utilizam por todas as zonas.

[g] YYMMDD – mais utilizado.

[h] É frequente encontar números com 6 zeros entre o número e a versão (Ex. 100000015).

[i] YYMMDDVV – mais utilizado.

[j] YYYYMMDDVV – mais utilizado.

[k] YYYYMMDVVV – mais utilizado.

348	deins.informatik.uni-dortmund.de
261	layon.inria.fr
133	noc.belwue.de
118	sun.iris-dcp.es
92	goya.dit.upm.es
88	aun.uninett.no
86	ns.germany.eu.net
86	nac.no
72	columba.udac.uu.se
68	runix.runit.sintef.no
64	hydra.helsinki.fi
63	shiva.jussieu.fr
62	cendrillon.lptl.jussieu.fr
57	rs2.hrz.th-darmstadt.de
57	rs1.hrz.th-darmstadt.de
55	cancer.ucs.ed.ac.uk
54	dns.dcs.ed.ac.uk
52	xlab-0.ed.ac.uk
52	wilde.ucs.ed.ac.uk
52	pythia.edvz.univie.ac.at
52	directory.ed.ac.uk
52	alf.uib.no

Tabela 5.4: Os 20 servidores mais populares na Europa (14/4/1992)

curiosa. De facto pode muito bem ter sido provocado pelo administrador de "cineca.it" que ao pretender comentar este RR no *master file* inseriu o caracter "#" no início da linha, em vez do ";" como deveria ser.

Na Irlanda parece que é popular atribuir outros nomes ao *backbone* da IE-Unet. Para além do verdadeiro nome, "dec4ie.ieunet.ie" existem ainda os: "dec4ie.eu.net" (com 19 referências), "dec4ie.ieunet.net" e "dec41e.ieunet.ie" (com uma referência cada um).

Uma das maiores causas de inconsistências no DNS são as referências de máquinas inválidas, principalmente provocadas pelas CNAME RRs mas também pelos MX RRs.

Existe um grupo de erros que são em parte provocados por uma certa ignorância dos administradores para com o significado de alguns RRs. Um exemplo disso é a insistência em utilizar o HINFO RR para associar informação a todo o tipo de nomes, como são os casos de:

```
uit.no.                          IN  HINFO   ''University'' ''ofTromso''
localhost.rz.uni-karlsruhe.de.   IN  HINFO   ''local'' ''local''
```

Na Tabela 5.6 são apresentados os 40 nomes de máquinas mais populares na Europa.

5627	mcsun.eu.net
5585	chenas.inria.fr
3800	dxmint.cern.ch
3654	cernvax.cern.ch
3266	sicvaxb.epfl.ch
3250	mail.germany.eu.net
2737	nsfnet-relay.ac.uk
2716	sun2.nsfnet-relay.ac.uk
2009	relay.surfnet.nl
1804	hearnvax.nic.surfnet.nl
1516	mailgzrz.tu-berlin.de
1419	mailszrz.zrz.tu-berlin.de
1379	nz11.rz.uni-karlsruhe.de
1379	netserv.rz.uni-karlsruhe.de
1335	sunic.sunet.se
1264	sun4nl.nluug.nl
1159	dutrun.tudelft.nl
1157	tudrva.tudelft.nl
1086	funet.fi
1059	chx400.switch.ch

Tabela 5.5: Os 20 *mail-gateways* europeus mais populares (14/4/1992)

Como seria de esperar, "pc1", "pc2", "pc3", ... , são muito populares. Curiosamente, "mac2" é mais popular que "mac1". Isto deve-se, muito provavelmente, ao facto de "mac2" definir também o modelo "Macintosh II".

Outro ponto importante é a presença de "localhost" como o nome mais popular na Europa (e provavelmente fora dela). "localhost", assim como "loopback" são nomes reservados para permitirem que qualquer máquina se possa designar a si próprio ou à sua rede. O primeiro está directamente relacionado com o endereço 127.0.0.1 e o segundo com o endereço 127 (correspondente à rede 127.0.0.0). Estes endereços denotam uma máquina virtual ("esta máquina") e uma rede virtual ("esta rede"). A sua introdução no DNS é desnecessária excepto para garantir a associação do nome "localhost" com o endereço 127.0.0.1 assim como a operação inversa ("reverse mapping").

A introdução do nome "localhost" do lado direito dos NS RRs ou MX RRs pode causar alguns efeitos secundários. No primeiro caso, leva a que qualquer máquina seja apresentada como servidor autoritário, enquanto no segundo provoca um *loop* no encaminhamento do *mail*, uma vez que, a utilização deste MX RR leva a que a mensagem seja entregue a si próprio.

1168	localhost	59	castor
117	venus	57	sirius
102	pluto	56	saturn
91	pc1	54	pc5
89	cisco	52	xterm1
88	mars	52	uranus
88	loopback	52	mac2
88	iris	51	pc6
85	pc2	51	helios
84	obelix	50	titan
80	asterix	50	gauss
79	idefix	48	snoopy
75	jupiter	48	bilbo
73	pc3	47	marvin
72	hermes	47	mac1
71	zeus	47	loghost
69	pollux	47	frodo
67	orion	47	donald
65	charon	47	apollo
61	pc4	46	merlin
		46	gandalf

Tabela 5.6: Top 40 dos nomes de máquinas na Europa (14/4/1992)

Capítulo 6

Conclusão

Os objectivos propostos no início do estágio foram, no essencial, cumpridos. A única alteração significativa foi o adiamento da montagem do *archive server* devido à falta de tempo. Esta falta de tempo ficou a dever-se, fundamentalmente, ao facto do estágio ter sido proposto para dois estagiários e não para um só, como veio a acontecer.

A construção de ferramentas de monitoragem da rede PTEUnet pode-se considerar um sucesso, assim como da base de dados de aderentes.

Actualmente a introdução ou modificação de um atributo de um aderente requer apenas alterações num único ficheiro (definição do aderente na base de dados) e a execução de um comando. Todas as repercussões nos ficheiros de administração são feitas automaticamente. Isto permite aumentar a eficiência do serviço prestado aos aderentes, uma vez que, os erros introduzidos inadvertidamente passaram a ser quase nulos (e centralizados num único ficheiro) e o tempo de resposta a pedidos deste tipo foi reduzido significativamente. Antes disso, a introdução de um novo aderente exigia um esforço maior. Como tarefas deste tipo não eram muito frequentes, levava a que nem sempre se tivesse presente quais os ficheiro que se deveria alterar e que alterações introduzir. Por outro lado, levava à introdução de informação redundante, uma vez que as fronteiras entre os diversos tipos de aderentes (UUCP sobre linha telefónica, TCP/IP, etc) não estavam bem definidas. A construção da base de dados exigiu uma análise exaustiva do funcionamento dos diversos sub-sistemas o que permitiu desenvolver um conjunto de *scripts* que manipulam os diversos ficheiros de administração. Assim passou a ser transparente para o administrador quais os ficheiros que na realidade são alterados e que alterações são introduzidas.

Também a monitorização do *backbone* foi simplificada com a realização dos programas de *accounting*. A sua utilização permite verificar o estado do *backbone* de uma forma bastante rápida e eficaz. Antes disso era necessário analisar os diversos *logs* para ter essa visão geral. A consulta dos dados devolvidos por estes programas, permite detectar e corrigir eventuais anomalias quando estas

ainda estão no início. Sem estas ferramentas muitas delas só seriam detectadas quando inviabilizassem ou detriorassem a prestação do serviço em questão.

Por outro lado, a experiência e os contactos que estabeleci durante o estágio permitiu-me colmatar algumas lacunas dos 4 primeiros anos da licenciatura. De facto, a experiência adquirida na administração de sistemas, ao longo do estágio, permite-me dispôr de um certo à vontade perante este tipo de tarefas. Para isso muito contribuiu o contacto que tive diariamente com a administração do *backbone* da PTEUnet, assim como as diversas análises que fiz ao longo de estágio (com destaque para a análise necessária para a construção da base de dados e que me permitiu ter uma visão de conjunto dos diversos sub-sistemas). Um outro aspecto importante foi a grande ênfase dada à elaboração de documentação, principalmente a documentação relativa ao DDT (onde mais de um terço do tempo foi dispendido na elaboração dos manuais para os diversos comandos assim como de um artigo com a sua apresentação) e a este relatório.

Normalmente um estágio deste tipo (profissionalizante) requer uma primeira fase para a inserção no local de trabalho e ambientação com o projecto. Neste caso concreto, isto não foi necessário, uma vez que eu já conhecia o ambiente de trabalho.

Capítulo 7

Agradecimentos

Agradecimentos são devidos ao Prof. Legatheaux Martins com quem o *feedback* mantido durante o estágio foi bastante importante, tanto na análise dos projectos como na elaboração da sua documentação, o que contribuiu em muito para o seu sucesso.

Bibliografia

[1] Eric Allman. *SENDMAIL Instalation and Operation Guide.* Britton-Lee, Inc. version 5.11++.

[2] J. M. Bloom and K. J. Dunlap. A Distributed Name Server for the DARPA Internet. *Usenix Conference Proceedings*, Summer 1986.

[3] Douglas E. Comer. *Internetworking with TCP/IP, Principles, Protocols and Architecture*, volume I. Prentice - Hall International, second edition, 1991.

[4] Douglas E. Comer and Thomas Narten. *UNIX Networking*, chapter TCP/IP. Hayden Books Unix System Library, 1990.

[5] Alan Emtage and Peter Deutsh. *archie - An Electronic Directory Service for the Internet.* McGill University, Montréal, Canada.

[6] Don Libes. expect: Curing Those Uncontrollable Fits of Interaction. In *Summer 1990 USENIX Conference*, Anaheim, California, June 10-15 1990.

[7] Don Libes. Using expect to Automate Systems Administration Tasks. In *Fourth USENIX Large Installation Systems Administration (LISA) Conference*, Colorado Springs, Colorado, October 17-19 1990.

[8] Don Libes. expect: Scripts for Controlling Interactive Processes. *Computing Systems*, 4(2), September 1991.

[9] M. Lottor. *RFC-1033 - Domain Administrators Operations Guide.* SRI International, November 1987.

[10] J. Legatheaux Martins. Relatório de execução do Projecto "Serviço IP da RCCN" durante o ano de 1991. Relatório FCUL-DI-92/03, Fundação para o Desenvolvimento dos Meios de Cálculo Científicos Nacionais, 1992.

[11] P. Mockapetris. *RFC-1034 - Domain Names - Concepts and Facilities.* USC/Information Sciences Institute, November 1987.

[12] P. Mockapetris. *RFC-1035 – Domain Names - Implementations and Specification.* USC/Information Scineces Institute, November 1987.

[13] Evi Nemeth, Garth Snyder, and Scott Seebass. *UNIX System Administration Handbook.* Prentice Hall, 1989.

[14] B. Clifford Neuman. *The Prospero File System User's Manual.* Department of Computer Science and Engineering, University of Washington, Seattle, Washington, 1991. Version Alpha.4.4.

[15] Tim O'Reilly and Grace Todino. *Managing UUCP and Usenet.* O'Reilly & Associates, Inc., 1989.

[16] C. Patridge. *RFC-974 – Mail routing and the domain system.* CSNET CIC BBN Labs, January 1986.

[17] Ian Lance Taylor. *Taylor UUCP*, 1992. version 1.02.

[18] Rebecca Thomas and Rik Farrow. *UNIX Administration Guide for System V.* Prentice Hall, 1989.

[19] Grace Todino and Dale Dougherty. *Using UUCP and Usenet.* O'Reilly & Associates, Inc., 1989.

Apêndice A

Relatório diário do backbone dec4pt

```
Sat May 16 07:00:02 WET DST 1992

-dec4pt.puug.pt- disk space:

Filesystem   Total    kbytes   kbytes   %
node         kbytes   used     free     used  Mounted on
/dev/rz1a      15823   12256     1985    86%  /
/dev/rz1d     140015  107019    27396    80%  /usr
/dev/rz1e      95983   19280    72864    21%  /home
/dev/rz1h     191983   36054   136731    21%  /mnt
/dev/rz1g     191983   79281   105023    43%  /usr/spool
/dev/rz1f     237168   14693   198759     7%  /mnt1
/dev/rz0a     396255  192248   164382    54%  /usr/spool/news
/dev/rz0d     552922  210522   287108    42%  /public


Sat May 16 07:00:03 WET DST 1992

-dec4pt.puug.pt- mail queue:

Mail queue is empty



Sat May 16 07:00:05 WET DST 1992

-dec4pt.puug.pt- sendmail report (last 24 hours)
```

Who	Class	Traffic	
apie	pt	17 messages	41859 bytes
	TOTAL	17 messages	41859 bytes
cprm	inter	1 message	1071 bytes
	TOTAL	1 message	1071 bytes
dns	pt	5 messages	8100 bytes
	TOTAL	5 messages	8100 bytes
fccn	pt	8 messages	9798 bytes
	TOTAL	8 messages	9798 bytes
inesc	pt	7 messages	17391 bytes
	TOTAL	7 messages	17391 bytes
ingrlis	inter	3 messages	8610 bytes
	pt	3 messages	3693 bytes
	TOTAL	6 messages	12303 bytes
isel	pt	3 messages	132255 bytes
	TOTAL	3 messages	132255 bytes
novabase	pt	1 message	34 bytes
	TOTAL	1 message	34 bytes
pt	pt	4 messages	1094 bytes
	TOTAL	4 messages	1094 bytes
puug	inter	43 messages	58220 bytes
	local	11 messages	5404 bytes
	pt	31 messages	16477 bytes
	TOTAL	85 messages	80101 bytes

```
saber-si        inter       3 messages      3188 bytes
                pt          2 messages       339 bytes
                -----   ------------------------------
                TOTAL       5 messages      3527 bytes

softlog         inter       5 messages      8143 bytes
                -----   ------------------------------
                TOTAL       5 messages      8143 bytes

ssf             inter     126 messages   1238588 bytes
                pt          9 messages      7618 bytes
                -----   ------------------------------
                TOTAL     135 messages   1246206 bytes

ua              pt         52 messages     85460 bytes
                -----   ------------------------------
                TOTAL      52 messages     85460 bytes

ul              inter      85 messages    125073 bytes
                pt         15 messages     10698 bytes
                -----   ------------------------------
                TOTAL     100 messages    135771 bytes

uminho          pt          8 messages     38071 bytes
                -----   ------------------------------
                TOTAL       8 messages     38071 bytes

unl             inter      32 messages     69971 bytes
                pt          1 message       9839 bytes
                -----   ------------------------------
                TOTAL      33 messages     79810 bytes

up              pt          2 messages      1871 bytes
                -----   ------------------------------
                TOTAL       2 messages      1871 bytes

uportu          pt          3 messages       183 bytes
                -----   ------------------------------
                TOTAL       3 messages       183 bytes


                local      11 messages      5404 bytes
                pt        171 messages    384780 bytes
                inter     298 messages   1512864 bytes
                =====   ==============================
                TOTAL     480 messages   1903048 bytes
```

```
-status statistics:

    1  Deferred: timeout waiting for input
    1  Deferred: Connection refused by muttley.fc.ul.pt
    1  Deferred: Network is unreachable
  482  Sent
    1  Deferred: Host is unreachable
    2  User unknown
    4  Host unknown


-mailers statistics:

   41  local
  243  TCP
  139  UUCP
   59  UUCP-A


-error statistics:

    1  SYSERR:=putbody:
   14  arpatounix:=unparseable
    1  Connecting=to
    8  unparseable,=received



Sat May 16 07:28:26 WET DST 1992

-dec4pt.puug.pt- Log requests for internet services (last 24 hours)


-ftp requests:

    1  192.68.221.60
    1  192.80.20.9
    2  rijp.ripe.net
    1  192.101.118.2
    1  corton.inria.fr
```

```
-finger requests:

    1  192.68.221.60


-telnet requests:

    1  mac-di-2.fc.ul.pt
    7  cptpuug.puug.pt
    1  mac-di-4.fc.ul.pt
    1  192.101.118.2


-rlogin requests:

    2  colombo
    4  muttley.fc.ul.pt
    1  inesc.inesc.pt


-nntp requests:

    7  colombo



Sat May 16 07:28:42 WET DST 1992

-dec4pt.puug.pt- DNS queries (last 24 hours)

Total queries received:  11200


Part I -- query sources -- top 20

Number  Source
------  ------
  8606 127.0.0.1
   556 192.67.76.5
   476 192.67.76.4
   156 192.33.4.12
   143 128.52.32.80
   109 129.79.1.9
    97 128.83.1.33
```

```
    46 130.136.1.6
    43 146.193.0.1
    39 131.114.3.163
    36 192.58.206.2
    36 16.1.240.15
    35 146.50.4.20
    31 192.36.148.17
    29 192.84.62.1
    28 133.4.11.2
    27 143.107.43.1
    26 192.122.238.22
    25 128.84.254.190
    17 192.82.214.19

Part II -- queried names -- top 20

Number  Queried name
------  ------------
   457 ssf.pt
   455 pandora.inesc.pt.inesc.pt
   424 fct.unl.pt
   409 di-fcul.fc.ul.pt
   396 ci.ua.pt
   346 journal.math.indiana.edu.puug.pt
   346 journal.math.indiana.edu
   296 di-fcul.fc.ul.pt.puug.pt
   216 di-fcul.fc.ul.pt.pt
   214 Pa.dec.com
   210 Pa.dec.com.puug.pt
   209 colombo.puug.pt
   173 ssf.pt.puug.pt
   167 colombo.puug.pt.puug.pt
   159 iuvax.cs.indiana.edu
   156 ci.ua.pt.puug.pt
   151 dec4pt.puug.pt.puug.pt
   147 inesc.inesc.pt
   120 iuvax.cs.indiana.edu.puug.pt
   114 localhost.pt

Part III -- query types

Number  Type
------  ----
  8450 A
  1589 MX
   456 PTR
   447 CNAME
```

```
   210 ANY
    41 SOA
     4 NS
     3 AXFR



Sat May 16 07:29:13 WET DST 1992

-dec4pt.puug.pt- UUCP report (last 24 hours)

 -spooldirs:

inesc     0C (0 secs)  15X (17 mins) TALKING              (05/16-07:10)
unl     420C (7 hours)  0X (0 secs)  CONVERSATION COMPLETE (05/15-23:07)
apie     30C (14 hours) 0X (0 secs)  CONVERSATION COMPLETE (05/15-14:56)
ssf     106C (1 hour)   0X (0 secs)  CONVERSATION COMPLETE (05/16-06:12)
softlog 86C (3 days)    0X (0 secs)  CONVERSATION COMPLETE (05/12-09:27)
nb        4C (17 hours) 0X (0 secs)


 -data sent statistics:

      files    bytes      secs    bytes/secs  baudrate
apie             11     13702         38        360      3605
inesc             0         0          0          0         0
ingrlis          34     71285        401        177      1777
isel             14    147080       1256        117      1171
nb                0         0          0          0         0
saber            10      7124         19        374      3749
softlog           0         0          0          0         0
ssf             430   6874862       9792        702      7020
unl             472  11739697      11795        995      9953
-------     -------   -------    -------    -------   -------
TOTAL           971  18853750      23301        809      8091


 -data received statistics:

      files    bytes      secs    bytes/secs  baudrate
apie             0         0          0          0         0
inesc          825  23946795      28301        846      8461
ingrlis          6      4362         27        161      1615
isel             8      2469         20        123      1234
```

```
nb              0         0         0         0         0
saber           6      1914         7       273      2734
softlog         0         0         0         0         0
ssf            34    158086       252       627      6273
unl             0         0         0         0         0
-------   -------   -------   -------   -------   -------
TOTAL         879  24113626     28607       842      8429


 -calls statictics:

in    out errors     retrans. (packets)
apie            4         0         1      0.00%
inesc           1         8         2      0.00%
ingrlis         7         0         0      0.00%
isel            7         0         3      2.06%
nb              0         0         0       --
saber           7         0         0      0.00%
softlog         0         0         0       --
ssf            10         0         2      0.04%
unl             9         0         3      0.13%
-------     ------    ------    ------     --------
TOTAL          45         8        11      0.07%


saber          7 (00:03) -  0.2%
ssf           10 (03:03) - 12.7%
isel          12 (00:30) -  2.1%
ingrlis        7 (00:10) -  0.7%
unl            9 (03:40) - 15.3%
apie           4 (00:02) -  0.1%
inesc          1 (00:00) -  0.0%
-------       ----------   -----
TOTAL         50 (07:28) -  4.4%


 -ttys statistics:

ttyd1        35 (06:40) - 27.8%
ttyd2        10 (00:47) -  3.3%
ttyp4         3 (00:01) -  0.0%
ttyp2         1 (00:00) -  0.0%
ttyp0         1 (00:00) -  0.0%
-------        ----------
TOTAL        50 (07:28) -  6.2%
```

Apêndice B

Lista de atributos de uma entidade

customer	no/mail/news/InterEUnet
remarks	comentario
uucpname	nome UUCP
country	pais da organizacao
machine	hardware;software
organization	nome da organizacao
person	nome do(s) contacto(s)
email	endereco de mail do(s) contacto(s)
telephone	telefone do(s) contacto(s)
address	endereco da organizacao
location	latitude / longitude
neighbors	sites de news adjacentes
pathalias	connectividades fisicas existentes
mailer	mailer utilizado quando ha' routing estatico
mail-relay	maquina para a qual e' feito routing estatico
mail-fee	classe de mail
transport	tipo de transporte utilizado
x25-map	endereco do router + endereco X.121
ip-route	endereco da rede do aderente + endereco do router
primary	somos primario deste dominio
secondary	somos secundario deste dominio
host	maquinas relevantes do aderente a inserir no seu dominio
bind-db	tipo de conectividade
news-fee	classe de news
news-transport	forma de difusao das news
news-cron	indicacao de quando processar as news
news-sys	entrada no ficheiro sys (news)
news-nntp	entrada no ficheiro nntp-access (nntp)
introduced	quem e quando introduziu esta entidade

Apêndice C

Definição sintáctica dos nomes em notação DNS

```
<dominio>       ::= <sub-dominio> | '' ''

<sub-dominio>   ::= <label> | <sub-dominio> ''.''  <label>

<label>         ::= <letra> [ [ <ldh-str> ] <let-dig> ]

<ldi-str>       ::= <let-dig-hif> | <let-dig-hif> <ldh-str>

<let-dig-hif>   ::= <let-dig> | ''-''

<let-dig>       ::= <letra> | <digito>

<letra>         ::= ''A'' | ...  | ''Z'' | ''a'' | ...  | ''z''

<digito>        ::= ''0'' | ...  | ''9''
```

Note-se que é indiferente se o nome é constituido por letras maiúsculas ou minúsculas.

Apêndice D

Entrega de uma mensagem de mail

O sistema DNS guarda informação necessária para definir políticas de encaminhamento de correio SMTP. Esta informação encontra-se codificada nos MX RRs. Estes *records* são constituídos por duas partes: um inteiro que indica a preferência com deve ser utilizado e o *host* para quem devem ser enviadas as mensagens. Isto permite que cada organização defina as suas próprias políticas de *routing* e proceda à sua alterações alterando apenas os dados relativos à sua zona.

Para além dos MX RRs por vezes são também utilizados os CNAME e WKS RRs. Os primeiros quando um nome, para quem é enviada a mensagem, é um alias para outro nome e o segundo indica se o *host* aceita ou não conexões SMTP (Simple Mail Transport Protocol), ou seja, se pode ou não receber o *mail* através de SMTP.

Como entregar a mensagem ?

Supondo que o *host* LOCAL tem uma mensagem endereçada a REMOTO (ambos os nomes estão sintaticamente correctos)[16].

O primeiro passo a fazer por LOCAL é tentar obter os MX RRs para REMOTO. A resposta por este obtida é considerada erro se:

- Não foi obtida nenhuma resposta. É importante distinguir a ocorrência de um erro da não obtenção de resposta.
- A resposta obtida está truncada. Nesta situação deve repetir a *query* utilizando um circuito virtual em vez de datagramas. A não aceitação desta resposta advém do facto de em certas circunstâncias a informação parcial poder causar ciclos.

- Ocorreu um erro (o campo da mensagem `Opcode` é diferente de zero).

Em caso de erro o comportamente a seguir pelos *mailers*[1] não está especificado. No entanto, este comportamente depende da natureza do erro. Se for um erro no contacto com o servidor, a mensagem poderá ser guardada para tentar mais tarde. Caso a origem do erro seja nome inexistente, a mensagem deverá ser imediatemente devolvida à precedência.

Se foi obtido um CNAME RR, isto é, REMOTO é um alias, a *query* deve ser repetida com o nome canónico e o processo voltará ao início.

Chegando a este ponto tem-se um lista de MX RRs. Se a lista for vazia (não existir nenhum MX RR para esse nome) é incluído na lista um MX RR a apontar para REMOTO com preferência nula. Isto permite que a mensagem seja entregue directamente ao *host* REMOTO, caso se trate de um *host*. Se não for o caso, a mensagem será devolvida aquando da obtenção do respectivo endereço. Caso a lista não seja vazia são retirados da lista os RRs irrelevantes de acordo com os passos seguintes:

- Todos os MX RRs em que o *host* para que apontam não disponibilize o serviço SMTP. Para o verificar são utilizados os WKS RRs. Na maioria dos casos esta verificação ainda não é feita.

- Se LOCAL pertencer à lista, todos os MX RRs com preferência maior ou igual à deste são retirados. Com isto pretende-se eliminar possíveis ciclos.

Se depois de procedida a esta remoção a lista ficar vazia, então entramos numa condição de erro. Normalmente nesta situação a mensagem é devolvida com uma frase indicando que não é possível proceder à sua entrega ("Undeliverable message!").

Finalmente, se a lista não estiver vazia o *mailer* tenta entregar a mensagem, começando pelo MX RRs de menor prioridade. Se não conseguir entregar ao primeiro tenta o seguinte, e assim sucessivamente até conseguir entregar ou ter percorrido toda a lista. Sempre que existam vários MX RRs com a mesma prioridade, a ordem por que serão utilizados é aleatória, estando apenas especificado que devem ser tentados todos antes de tentar um com preferência superior. Normalmente, o *mailer* depois de tentar todos os RRs não devolve a mensagem de imediato, guarda-a para tentar a sua entrega mais tarde. Na próxima tentativa recomeçará o processo pela obtenção dos MX RRs.

Questões importantes

Muitas vezes são definidos MX RRs para nomes *wildcards*, ou seja, nomes do tipo "*.domain". Em situações destas é necessário tomar alguns cuidados na sua utilização, uma vez que o seu significado é muitas vezes mal interpretado. De facto, é necessário ter em conta que estes só são utilizados quando não existir

[1] Programa que procede à entrega da mensagem.

nenhuma informação associada ao nome em causa. Supondo por exemplo a existência dos seguintes RRs:

```
*.ul.pt.          MX  10  muttley.ul.pt.
muttley.ul.pt.    A   192.67.76.5
fc.ul.pt.         MX  10  di-fcul.ul.pt.
```

O *wildcard* nunca será utilizado para nomes como: "muttley.ul.pt.", "fc.ul.pt.", "ptearn.fc.ul.pt.", etc. Por outro lado, para o nome "bach.ul.py." já será utilizado. A existência de nomes deste tipo é transparente para o *mailer* uma vez que o servidor devolverá unicamente um MX associado a esse nome e não o próprio *wildcard*. No exemplo anterior o servidor nunca indicaria a existência do MX RR para "*.ul.pt" se lhe perguntassem pelos MX RRs assiciados ao nome "bach.ul.pt", mas sim o MX RR para bach.ul.pt a apontar para "muttley.ul.pt" com uma preferência de 10.

A definição de MX RRs a apontar para aliases pode causar alguns problemas. Isto porque o algoritmo que retira da lista todos os MX RRs com prioridade maior ou igual à do que aponta para local (caso este exista, obviamente) não verifica se este está definido como um *nickname* (alias). O que pode vir a causar um ciclo, como foi referido atrás.

Apêndice E

Códigos ISO 3166 para os países

AT -- Austria
BE -- Bélgica
BG -- Bulgária
CH -- Suiça
CS -- Checoslováquia
DE -- Alemanha
DK -- Dinamarca
ES -- Espanha
FI -- Filândia
FR -- França
GR -- Grécia
HU -- Hungria
IE -- Republica da Irlanda
IL -- Israel
IS -- Islandia
IT -- Itália
NL -- Holanda
NO -- Noruega
PL -- Polónia
PT -- Portugal
SE -- Suécia
SU -- CEI (antiga União Soviética)
TN -- Tunísia
UK -- Reino Unido
YU -- Jugoslávia

Lista de Figuras

Lista de Tabelas

Índice