Laboratórios de Informática 2014/2015
DETI - Universidade de Aveiro

Docentes
João Paulo Barraca <jpbarraca@ua.pt>
André Zúquete <andre.zuquete@ua.pt>
Bernardo Cunha <mbc@ua.pt>

TEMA 4

Redes de Comunicações

Objetivos:
- Conceito de endereço IP
- Máscaras
- Rotas
- Configuração de rede em Linux
- Serviços de Rede
- Acesso Remoto

4.1 Introdução

Os sistemas com capacidades de comunicação em rede possuem uma variedade de identificadores que possibilitam a troca de informação. Estes identificadores funcionam como a morada numa casa, quando se pretende trocar correspondência, ou o número de telefone quando se pretende falar com um amigo. Em ambos os casos existem identificadores que permitem que a informação chegue ao seu destino, e se identifique a origem.

Neste trabalho iremos explorar como estes identificadores estão relacionados e qual a sua utilidade para a comunicação na Internet.

Recomenda-se a utilização da máquina virtual disponibilizada para a disciplina.

Importante: Antes de iniciar este guia, cria uma máquina virtual através do disco fornecido, mas crie 2 interfaces de rede. O primeiro do tipo *NAT*[1], e servirá para

[1] ver [1] ou simplesmente `http://en.wikipedia.org/wiki/Network_address_translation`

comunicação com a Internet. O outro interface será do tipo *Internal Network* e servirá para comunicação entre outras máquinas virtuais.

4.2 Configuração de rede de um PC

A configuração de rede de um PC considera diversos componentes. Os mais importantes são: endereços, interfaces, encaminhamento e serviço de resolução de nomes. De seguida iremos abordar a configuração de alguns parâmetros.

Exercício 4.1

Abra um terminal, execute o comando `ifconfig` e verifique:

1. Quantos interfaces de rede existem;
2. O endereço Internet Protocol v4 (IPv4)[2] de cada interface;
3. O(s) endereço(s) Internet Protocol v6 (IPv6)[3] de cada interface;
4. O endereço físico (Media Access Control (MAC)[4]) de cada interface;
5. O tipo de cada interface;
6. A máscara de rede de cada interface;
7. Relacione o número de interfaces reportados no *Linux*, com o número reportado pelo *VirtualBox*.

Exercício 4.2

De forma a determinar as rotas existentes, execute `route -n`. Verifique qual a rota por omissão (*default*) do sistema.

Também encontra outras rotas, quais?

Os sistemas *Linux* necessitam de informação acerca da configuração dos interface de rede. Esta configuração permite especificar se os interfaces são para ser configurados usando algum método dinâmico (ex, Dynamic Host Configuration Protocol (DHCP)[5]), ou através de uma configuração estática. Pode verificar a configuração acedendo ao ficheiro `/etc/network/interfaces`.

Como exemplo, considere a seguinte configuração que define que o interface `eth2` será configurado dinamicamente, enquanto o `eth3` será configurado estaticamente:

```
auto eth2
iface eth2 inet dhcp

auto eth3
iface eth3 inet static
    address 192.168.0.1
    netmask 255.255.255.0
    gateway 192.168.0.254
```

Exercício 4.3

Edite o ficheiro `/etc/network/interfaces` e aplique uma configuração de rede estática de forma a configurar o primeiro interface de rede (*NAT*) como sendo dinâmico, e o segundo interface de rede (*Internal Network*) como estático. Para esta configuração, considere uma rede 192.168.56.0/24 e sem gateway.

Pode aplicar a configuração através dos comando `ifup nome-do-interface`. O comando `ifdown nome-do-interface` permite desactivar a configuração. E o comando `ifconfig nome-do-interface` permite verificar o estado atual.

Através do interface gráfico também é possível realizar as mesmas configurações, mas de forma gráfica. No caso do *Lubuntu* é disponibilizada uma aplicação no menu *Preferências/Ligações de Rede*.

Exercício 4.4

Utilize a interface gráfica para aplicar uma configuração que define todos os interfaces como utilizando endereços obtidos dinamicamente (DHCP).

4.3 Tabela de Endereços Físicos

Os dispositivos com capacidade de comunicação possuem endereços únicos que os identificam. O sistema operativo mantém uma tabela onde regista informação sobre as estações **vizinhas conhecidas**. Em *Linux* é possível listar as entradas desta tabela executando o

comando `arp -a`. A Figura 4.1 demonstra a tabela que pode encontrar.

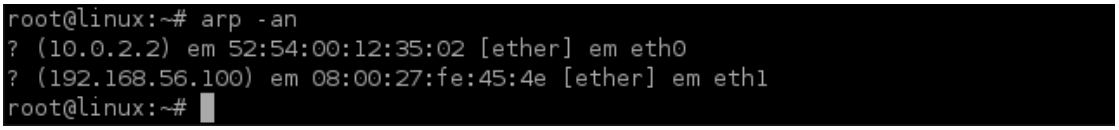

Figura 4.1: Resultado do comando codearp -an.

Exercício 4.5

Execute o comando `arp -an`, verifique se o endereço IPv4 de um Computador Pessoal (PC) ao seu lado está presente na tabela (terá de o perguntar ao colega que está nesse computador).

Repita o comando no PC da sala de aula e compare resultado.

Exercício 4.6

De seguida, execute um `ping endereço-de-destino` para o endereço do PC ao seu lado e volte a observar o conteúdo da tabela. Este comando também funciona dentro da máquina virtual?

Exercício 4.7

Utilizando o PC da sala de aula, repita o comando `ping endereço-de-destino` Espere 3 minutos sem gerar qualquer tráfego e volte a observar o conteúdo da tabela, identifica alguma alteração?

4.4 Tradução de nomes em endereços IP

Os nomes que utilizamos para aceder a conteúdos HTTP não são os utilizados para as comunicações. Na realidade o endereço IPv4 (v4 ou v6) é que é utilizado a quando do estabelecimento de uma ligação. Existe um serviço que permite traduzir nomes (ex. `www.ua.pt`) em endereços IPv4, e vice versa.

Exemplos de alguns nomes:

```
www.ua.pt
www.up.pt
www.sapo.pt
www.antena3.pt
www.fcporto.pt
www.scp.pt
www.sporting.pt
www.slbenfica.pt
www.google.com
www.google.pt
www.facebook.com
```

Exercício 4.8

A configuração de Dynamic Name System (DNS)[6] de um sistema *Linux* encontra-se em `/etc/resolv.conf`. Visualize o conteúdo deste ficheiro e registe qual o servidor de DNS que está a utilizar.

Compare este valor com o presente no anfitrião da máquina virtual.

Exercício 4.9

Para os nomes anteriormente listados, e utilizando o comando `host` (ex., `host www.ua.pt`), registe qual(ais) os endereços associados (resolução direta). Seja curioso. Procure e registe endereços repetidos, múltiplos endereços ou outras situações que considere anómalas.

Exercício 4.10

Da mesma forma que é possível traduzir nomes em endereços, também é possível realizar a operação inversa. Utilizando o mesmo comando (ex., `host 193.136.92.123`), verifique qual a correspondência inversa (de endereço para nome). Procure identificar se a resolução direta e inversa produzem resultados compatíveis.

4.5 Conectividade e rotas

Até agora sabemos que é possível comunicar dado que se sabe o endereço IPv4 do servidor. Também já foi abordado o serviço que permite converter nomes em endereços IPv4. Resta saber como a informação atravessa a Internet. O segredo está no conceito de rota de encaminhamento, o que já verificou utilizando o comando `route`.

Existem dois comandos particularmente relevantes no domínio do diagnóstico do estado das redes e das sua rotas: `ping` e `traceroute`. O primeiro (`ping`) permite enviar um pacote especialmente construído que instrui o destinatário a responder. Pode ser utilizado para determinar a existência de conectividade e mesmo o atraso nas comunicações. O segundo comando (`traceroute`) é mais complexo, permitindo identificar a rota utilizada para comunicar com o destino.

Exercício 4.11

Execute o comando `ping` para cada um dos destinos apresentados e registe o tempo médio de comunicação. Pode também verificar que algumas ligações apresentam ocasionalmente perdas de pacotes. Detecta uma correlação entre tempo, perdas e distância?

Nome	Localização
www.ua.pt	Aveiro, Portugal
www.up.pt	Porto, Portugal
www.utl.pt	Lisboa, Portugal
www.utad.pt	Vila Real, Portugal
www.uevora.pt	Évora, Portugal
www.uam.es	Madrid, Espanha
www.univ-paris8.fr	Paris, França
www.cmu.edu	Pittsburgh, EUA
www.bjut.edu.cn	Pequim, China
www.u-tokyo.ac.jp	Tóquio, Japão
www.adelaide.edu.au	Adelaide, Austrália
www.cstome.net	São Tomé e Príncipe

Enviando pacotes especialmente construídos, e processando as notificações enviadas de volta por cada *Router*, é possível identificar os dispositivos numa rota. O programa `traceroute` implementa este mecanismo de sinalização. A Figura 4.2 demonstra a rota

que existe entre a Universidade de Aveiro e os servidores de `www.google.pt`. Para cada uma das entradas é mostrado o nome, endereço IPv4 e o tempo médio de resposta.

```
traceroute to www.google.pt (173.194.45.23), 30 hops max, 60 byte packets
 1  193.137.173.209 (193.137.173.209)  0.389 ms  0.665 ms  0.746 ms
 2  10.0.34.1 (10.0.34.1)  0.609 ms  0.607 ms  0.713 ms
 3  Router2.Campanha.fccn.pt (193.136.4.26)  0.930 ms  0.965 ms  0.954 ms
 4  Router3.10GE.DWDM.Lisboa.fccn.pt (193.136.1.1)  5.621 ms  5.686 ms *
 5  ROUTER10.10GE.CR1.Lisboa.fccn.pt (193.137.0.8)  5.480 ms  5.474 ms  5.458 ms
 6  Google.AS15169.gigapix.pt (193.136.250.20)  5.611 ms  5.616 ms  5.617 ms
 7  209.85.254.70 (209.85.254.70)  6.435 ms  6.503 ms  6.494 ms
 8  lis01s06-in-f23.1e100.net (173.194.45.23)  5.645 ms  5.640 ms  5.653 ms
```

Figura 4.2: Resultado do comando `traceroute www.google.pt`.

Devido às regras de segurança aplicadas na Universidade de Aveiro, não é possível utilizar o programa `traceroute` dentro da rede da universidade. Como tal, recomenda-se utilizar um serviço HyperText Transfer Protocol (HTTP)[7] que permite executar o comando `traceroute` remotamente. A origem dos pacotes será Portugal, mas não será Aveiro. Para aceder ao serviço, utilize o navegador que tem instalado e insira o endereço:

`http://glass.cprm.net/`

Pode experimentar vários endereços através deste interface. Se estiver a realizar este guia fora da Universidade de Aveiro, poderá utilizar o comando `traceroute endereço` directamente a partir da linha de comandos da máquina virtual.

Exercício 4.12

Para cada um dos endereços anteriormente analisados, obtenha a rota desde a o ponto de origem até ao destino. De seguida, analise a rota obtida, identifique e registe:

1. O número de cada *Router* da rota.
2. O número de países diferentes por onde o tráfego foi encaminhado.
3. O *Router* com maior atraso.

4.6 Identificação da entidade responsável por uma máquina

Todos os equipamentos possuem uma entidade responsável, e esta entidade tem de estar devidamente identificada perante os restantes utilizadores da Internet. Bases de dados disponíveis *online*, como por exemplo `http://www.whois.sc` permitem consultar esta informação.

Exercício 4.13

Para cada um dos nomes, registe o nome do titular do registo.

Exercício 4.14

Considerando as rotas obtidas anteriormente, e utilizando o serviço
`http://cqcounter.com/whois/`
registe qual a entidade responsável (Organization), pelo acesso à Internet de cada um dos destinos.

4.7 Transmissão de informação em redes: `ping`

Até agora tem-se referido que as redes atuais são orientadas à comunicação por pacotes, não tendo sido no entanto observados estes elementos de comunicação. Neste ponto iremos observar o que realmente acontece quando se realiza o comando `ping www.google.pt`.

Para isso é necessário instalar uma aplicação que permite escutar todo o tráfego enviado para a rede: *Wireshark*. Instale a aplicação na máquina virtual e seguidamente execute-a com permissões de super-utilizador.

De forma a capturar tráfego, efectue os seguintes passos:

- Aceda às opções do *Wireshark* e defina que quer escutar pacotes no interface de rede que configurou no *VirtualBox* como sendo *NAT*;
- Quais os endereços IPv4 e MAC envolvidos nas comunicações.
- Que protocolos são utilizados em cada comunicação.
- Consegue identificar o endereço do servidor de DNS?

· Como o comando `ping` consegue saber a que pergunta corresponde uma resposta?

Exercício 4.15

Repita o comando `ping` para vários endereços. Consegue explicar o funcionamento do comando?

4.8 Transmissão de informação em redes: conteúdo HTTP

O protocolo HTTP é um protocolo de nível aplicacional, muito utilizado para a transferência de informação na Internet. Sempre que acede ao *Google*, ou *Facebook* está a utilizar este protocolo. Visto ser um protocolo aplicacional, ele funciona em cima de um outro protocolo chamado Transmission Control Protocol (TCP)[8]. Este protocolo (TCP), permite que vários serviços o utilizem, criando a noção de portas. Cada comunicação usa uma porta diferente e assim é possível comunicar. Isto será abordado nos parágrafos seguintes.

Para transferir a informação, o protocolo HTTP baseia-se no princípio da pergunta e resposta. Quando insere um Uniform Resource Locator (URL)[9] no browser, é enviada uma pergunta, que é respondida com o conteúdo pretendido.

O exemplo que se segue é uma destas perguntas. Neste caso, questiona-se um servidor `www.google.pt` pela página `/`. Como pode verificar, o cliente também se identifica (`User-Agent`) e define que tipo de conteúdo aceita (`Accept`).

```
GET / HTTP/1.1
Host: www.google.pt
User-Agent: Mozilla/5.0
Accept: text/html
```

Exercício 4.16

O comando `telnet endereço-do-servidor porta` permite efetuar uma ligação TCP, sobre a qual se pode transmitir informação. Para verificar o quanto simples é o protocolo HTTP, efectue uma ligação ao servidor indicado na porta 80 (ex, `telnet www.google.pt 80`) e envie a pergunta mostrada anteriormente. Deverá aparecer muito texto.

Utilize o *Wireshark* e verifique o que realmente acontece.

Exercício 4.17

Utilizando o browser, repita o processo para qualquer outro site, e analise o resultado com o *Wireshark*.

Consegue identificar os endereços IP e os protocolos utilizados?

Relativamente ao protocolo HTTP, consegue identificar qual a versão do protocolo, cliente utilizado, servidor e caminho que compõem o URL pedido?

4.9 Acesso Remoto

4.9.1 SSH

Tal como já visto, é possível realizar ligações a sistemas e trabalhar nestes, como se o sistema local se tratasse. Um dos métodos mais utilizados nos dias de hoje é o `ssh`, acrónimo de Secure Shell. No passado, o método mais utilizado era o protocolo `telnet`, que ainda se pode encontrar em alguns equipamentos mais simples.

O `ssh` possibilita o acesso a sistemas remotos, invocando uma *shell* no sistema remoto, sobre a qual é possível executar comandos. Ao contrário do `telnet` toda a informação trocada através do `ssh` são seguras, não sendo possível a atacantes interceptarem e entenderem as acções do utilizador. Existem algumas outras funcionalidades que levaram a que o `ssh` se tornasse mais popular:

- Suporta níveis de segurança configuráveis.
- Suporta a transferência de ficheiros.
- Suporta a criação de túneis de tráfego (como uma Virtual Private Network (VPN)[10]).
- Suporta a execução de aplicações gráficas remotas.

Para iniciar uma ligação `ssh` é apenas necessário executar:

`ssh username@servidor` ou alternativamente `ssh -l username servidor`.

No caso desta disciplina, o servidor mais utilizado será `xcoa.av.it.pt` enquanto o username terá o formato `labi-tXgY`.

Portanto, ao executar na consola:

`ssh labi-tXgY@xcoa.av.it.pt`

, o grupo Y da turma X estará a iniciar uma sessão remota no servidor `xcoa.av.it.pt`. Use o seu número de grupo e turma.

Como mecanismo de segurança, o `ssh` cria uma impressão digital dos servidores. Isto permite a que os utilizadores tenham a certeza que se estão a ligar ao servidor certo[2] No caso do servidor atual, e na primeira ligação, será apresentada a seguinte mensagem:

```
The authenticity of host 'xcoa.av.it.pt (193.136.92.155)' can't be established.
ECDSA key fingerprint is 6d:ec:23:6a:ff:9a:7c:02:47:23:8f:82:83:2d:1c:23.
Are you sure you want to continue connecting (yes/no)?
```

De reparar que a impressão digital (*fingerprint*) do servidor tem o valor

`6d:ec:23:6a:ff:9a:7c:02:47:23:8f:82:83:2d:1c:23`.

Caso este valor seja apresentado, o utilizador sabe que está a ligar-se ao servidor correto. Caso seja diferente, deverá-se interromper imediatamente a tentativa de ligação.
Os valores das impressões digitais é armazenado em `~/.ssh/known_hosts`.

Numa ligação seguinte, nenhuma verificação é pedida ao utilizador. Caso a impressão digital guardada seja diferente da do servidor é mostrada a mensagem:

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that a host key has just been changed.
The fingerprint for the RSA key sent by the remote host is
ec:3c:30:76:b6:a7:b9:8c:17:73:25:da:10:e7:7f:03.
Please contact your system administrator.
Add correct host key in /home/linux/.ssh/known_hosts to get rid of this message.
Offending RSA key in /home/linux/.ssh/known_hosts:1
RSA host key for xcoa.av.it.pt has changed and you have requested strict checking.
Host key verification failed.
```

[2]Considere-se que é possível desviar as comunicações que passam na Internet, tal como um carteiro poderia desviar cartas se assim o entendesse.

Depois da sessão se encontrar estabelecida, todos os comandos introduzidos são executados no servidor remoto, sendo o seu resultado enviado de volta para o cliente local. Pode verificar que utilizadores se encontram ligados executando o comando `who`.

Exercício 4.18

Remova o ficheiro ∼`/.ssh/known_hosts` do seu sistema. De seguida efetue uma ligação ao servidor *xcoa.av.it.pt* e verifique que a impressão digital é a apresentada neste guião. Termine a ligação e volte a restabelece-la. É apresentada alguma mensagem adicional?

Verifique agora o conteúdo do ficheiro ∼`/.ssh/known_hosts` e localize a entrada relacionada com o servidor *xcoa.av.it.pt*.

4.9.2 Transferência de ficheiros

O `ssh` não permite apenas executar comandos remotos, permite igualmente transferir ficheiros entre sistemas. É possível transferir ficheiros e diretórios para servidores remotos, entre servidores remotos, ou obter ficheiros e diretórios para o computador local. Como exemplo, é possível transferir as páginas criadas para o servidor remoto, ou enviar imagens para melhorar a página.

Quando se pretende transferir um ficheiro utilizando `ssh`, deverá ser invocado o comando `scp` (Secure Copy). Este comando possui alguns parâmetros semelhantes ao `cp` que já foi utilizado em aulas passadas. A sintaxe do comando `scp` baseia-se no mesmo conceito do `cp` (`cp origem destino`) e é a seguinte:

```
scp utilizador@origem:directorio utilizador@destino:directorio
```

em que a *origem* ou *destino* indicam o servidor de origem e destino, sendo igualmente importante especificar o utilizador e o caminho. O seguinte exemplo copia um ficheiro *teste.txt*, que se encontra na área pessoal (*home*) do utilizador atual para a área pessoal do utilizador chamado *user* servidor *xcoa.av.it.pt*.

```
scp ~/teste.txt user@xcoa.av.it.pt:/home/user
```

Para copiar o ficheiro de volta, desta vez para o diretório atual, poderia ser executado:

```
scp user@xcoa.av.it.pt:/home/user/teste.txt .
```

Também é possível copiar o ficheiro entre sistemas remotos:

```
scp user1@xcoa.av.it.pt:/home/user/teste.txt user2@xcoa.av.it.pt:
```

Exercício 4.19

Usando o browser, obtenha uma imagem para o seu computador, e utilizando o `scp`, copie-a para o servidor *xcoa.av.it.pt*.

Autenticação por chaves

Até agora a autenticação do protocolo `ssh` junto de servidores remotos tem sido efetuada através de um nome de utilizador e de uma senha. Quando se estabelecem ligações frequentemente a servidores, o facto de se introduzirem constantemente estes dados torna-se problemático. Por outro lado, a utilização deste par de elementos (utilizador e senha), não é a mais segura. Alem de outros problemas de segurança, senhas curtas ou baseadas em palavras de dicionário podem fornecer a terceiros acesso ao acesso remoto.

O `ssh` permite a utilização de um par de chaves que irá substituir a senha. Estas chaves são constituídas por duas partes (2 ficheiros). Uma é pública e deve ser colocada nos servidores a que pretendemos ligar, a outra nunca deve ser fornecida a terceiros, ficando no computador local [3].

O primeiro passo a fazer é a criação das chaves para autenticação. A chave local deve estar cifrada com uma senha. Esta senha nunca é enviada para o servidor, serve apenas para decifrar o ficheiro `id_rsa`.

Isto consegue-se executando o comando `ssh-keygen` e seguindo as instruções fornecidas:

[3] O conceito de chaves públicas e privadas está fora do âmbito desta disciplina. Se tiver curiosidade pode consultar a página http://pt.wikipedia.org/wiki/Criptografia_de_chave _pública

```
\$> ssh-keygen Generating public/private rsa key pair.
Enter file in which to save the key (/home/user/.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/user/.ssh/id_rsa.
Your public key has been saved in /home/user/.ssh/id_rsa.pub.
The key fingerprint is:
cd:78:2d:63:12:aa:02:89:22:de:89:4d:cd:3d:8e:73 user@labi
```

A partir deste momento estarão criados dois ficheiros com as duas chaves:

- Chave Pública (a colocar no servidor): `/home/linux/.ssh/id_rsa.pub`
- Chave Privada (a manter localmente): `/home/linux/.ssh/id_rsa`

A chave pública deve ser adicionada ao ficheiro `~/.ssh/authorized_keys` para que o `ssh` passe a utilizar as chaves criadas em vez do método tradicional para autenticação.

Exercício 4.20

Crie um par de chaves no seu computador sem especificar uma palavra passe. Instale a chave pública na sua conta do servidor *xcoa.av.it.pt* e verifique o que acontece quanto volta a estabelecer uma sessão ao servidor.

Volte a criar e instalar um par de chaves mas especificando uma senha. Volte a estabelecer uma sessão ao servidor e verifique o que acontece.

Pode obter mais informação sobre o processo de autenticação se executar `ssh -v` em vez de `ssh`.

Reencaminhamento do protocolo X11

A execução de aplicações possuindo interface gráfico também é possível através de `ssh`. Em *Linux*, o interface gráfico é um serviço (do protocolo X11 [4]) que recebe pedidos de aplicações (ex., desenhar um botão, apresentar uma imagem). Como a integração é feita através de mensagens, torna-se assim possível que a aplicação e o interface gráfico da mesma se encontrem em sistemas distintos.

[4] Para mais informação, consultar `http://pt.wikipedia.org/wiki/X_Window_System`

Utilizando `ssh` existe a possibilidade de utilizar o servidor local para uso das aplicações remotas. Considerando o servidor `xcoa.av.it.pt`, aplicações com suporte gráfico que executem neste servidor podem apresentar o seu interface no computador local. Para isto é necessário executar o `ssh` da seguinte forma:

```
ssh -X user@servidor
```

Depois, qualquer aplicação gráfica poderá executar não estando as sessões remotas limitas a comandos de texto. Se se pretender iniciar uma sessão sem suporte de reencaminhamento de X11, o `ssh` poderá ser executado com a opção `-x` (minúsculo).

Exercício 4.21

Efetue uma ligação ao servidor *xcoa.av.it.pt* com a opção `-x` e interprete o resultado do comando `midori`.

Volte a repetir a sessão mas desta vez utilizando a opção `- X`. Compare o resultado que obtem ao executar o comando `midori`.

Utilizando o `midori` (no servidor remoto) e o `firefox` (no computador local) aceda ao URL `http://labi.aws.atnog.av.it.pt/ip/` e compare o resultado. Pode igualmente aceder a outras páginas como `http://my.ua.pt` ou `http://www.sapo.pt` e verificar que em ambos os casos existe conetividade à Internet.

4.10 Para aprofundar o tema

Exercício 4.22

Utilizando o wireshark, capture tráfego e identifique todos os pedidos efectuados. Pode utilizar o filtro `http.request` depois de capturar os pacotes se quiser visualizar apenas os pedidos de HTTP.

Exercício 4.23

Execute o comando `traceroute` e, através da aplicação *Wireshark*, verifique que pacotes são enviados. Tente encontrar a função do campo `TTL` e como este é utilizado.

Exercício 4.24

Execute o comando `ftp glua.ua.pt` e utilize o utilizador `ftp` com a password `ftp`. Capture o tráfego e verifique que dados consegue visualizar na captura.

Exercício 4.25

É comum nomear os sistemas com personagens e locais de livros, séries, filmes, sagas ou outras obras. Sabendo que os sistemas centrais da Universidade de Aveiro estão na rede `193.136.173.0/24`, e recorrendo ao comando `host` ou `dig -x`, resolva vários endereços IPv4 e determine qual(ais) as obras que são utilizadas para nomear alguns sistemas da universidade.

Glossário

DHCP Dynamic Host Configuration Protocol

DNS Dynamic Name System

HTTP HyperText Transfer Protocol

IPv4 Internet Protocol v4

IPv6 Internet Protocol v6

MAC Media Access Control

PC Computador Pessoal

TCP Transmission Control Protocol

URL Uniform Resource Locator

VPN Virtual Private Network

Referências

[1] K. Egevang e P. Francis, *The IP Network Address Translator (NAT)*, RFC 1631 (Informational), Obsoleted by RFC 3022, Internet Engineering Task Force, mai. de 1994.

[2] J. Postel, *Internet Protocol*, RFC 791 (Standard), Updated by RFC 1349, Internet Engineering Task Force, set. de 1981.

[3] S. Deering e R. Hinden, *Internet Protocol, Version 6 (IPv6) Specification*, RFC 2460 (Draft Standard), Updated by RFCs 5095, 5722, 5871, Internet Engineering Task Force, dez. de 1998.

[4] C. Hornig, *A Standard for the Transmission of IP Datagrams over Ethernet Networks*, RFC 894 (Standard), Internet Engineering Task Force, abr. de 1984.

[5] R. Droms, *Dynamic Host Configuration Protocol*, RFC 2131 (Draft Standard), Updated by RFCs 3396, 4361, 5494, Internet Engineering Task Force, mar. de 1997.

[6] P. Mockapetris, *Domain names - implementation and specification*, RFC 1035 (Standard), Updated by RFCs 1101, 1183, 1348, 1876, 1982, 1995, 1996, 2065, 2136, 2181, 2137, 2308, 2535, 2845, 3425, 3658, 4033, 4034, 4035, 4343, 5936, 5966, Internet Engineering Task Force, nov. de 1987.

[7] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach e T. Berners-Lee, *Hypertext Transfer Protocol – HTTP/1.1*, RFC 2616 (Draft Standard), Updated by RFCs 2817, 5785, 6266, Internet Engineering Task Force, jun. de 1999.

[8] J. Postel, *Transmission Control Protocol*, RFC 793 (Standard), Updated by RFCs 1122, 3168, 6093, Internet Engineering Task Force, set. de 1981.

[9] M. Mealling e R. Denenberg, *Report from the Joint W3C/IETF URI Planning Interest Group: Uniform Resource Identifiers (URIs), URLs, and Uniform Resource Names (URNs): Clarifications and Recommendations*, RFC 3305 (Informational), Internet Engineering Task Force, ago. de 2002.

[10] L. Andersson e T. Madsen, *Provider Provisioned Virtual Private Network (VPN) Terminology*, RFC 4026 (Informational), Internet Engineering Task Force, mar. de 2005.