

# GNS3 2.2.29 tutorial

---

## Índice

1	Introdução.....	3
2	Instalação do GNS3.....	4
3	Instalar imagem do IOS fornecida .....	5
3.1	Instalação da imagem.....	5
3.2	Informação acerca da imagem c2600-is-mz.123-17a.bin fornecida .....	6
4	GNS3 .....	8
4.1	Ecrã.....	8
4.2	Ficheiros.....	8
4.3	Gravação de um projecto .....	8
5	Uso do Wireshark no GNS3.....	9
5.1	Captura de pacotes no GNS3 .....	9
6	Comandos básicos a usar nos dispositivos.....	11
6.1	Dispositivos IOS.....	11
6.2	Virtual PC Simulator (VPCS ).....	12
7	Primeira simulação de uma rede no GNS3.....	14
7.1	Cenário a simular .....	14
7.2	Construção da rede.....	14
7.3	Configurações .....	15
7.4	Captura de pacotes usando o <i>Wireshark</i> .....	19
8	Containers no GNS3 .....	20
8.1	Instalação do Docker – exemplo do Linux (LUbuntu).....	20
8.2	Criação de um <i>container</i> .....	20
8.3	Instalação do <i>container</i> criado no GNS3 .....	21
9	Problemas de instalação .....	23

## 1 Introdução

Este tutorial pretende servir de apoio à parte prática da disciplina de **Redes de Comunicação (DEI-FCTUC)**, explicando as bases do funcionamento do *Graphical Network Simulator 3* (GNS3).



O GNS3 é um simulador gráfico de redes de computadores que permite emular vários sistemas tais como *routers* Cisco e Juniper, máquinas virtuais Linux e Windows. Um *router* é um sistema de rede responsável pelo encaminhamento (ou *forwarding*) de pacotes entre redes desempenhando, por conseguinte, um papel vital no funcionamento da Internet. No GNS3, os *routers* da Cisco são simulados usando o *software* de emulação *Dynamips*.

Ao contrário de outros simuladores de redes, o GNS3 emula o *hardware* existente, funcionalidade que lhe permite correr imagens reais do sistema operativo Cisco IOS (*software* que corre em equipamento de rede da Cisco). Ao usar estas imagens reais é possível usar todos os comandos existentes e obter os mesmos resultados de um ambiente real.

Além de emular *hardware*, o GNS3 também permite interligar diversos dispositivos incluindo outros emuladores, máquinas virtuais ou equipamento real.

### Convenções usadas no documento:

- Menus e opções em negrito itálico
  - “Aceda ao menu **View / Docs** e escolha o item **Console**”
- Comandos a digitar e *command line prompt*:  
**Router#** configure terminal
- Texto em destaque a **negrito** ou *itálico*

### Nota:

Se estiver a usar a máquina virtual fornecida na disciplina de Redes de Comunicação do DEI-FCTUC, o GNS3 e o Docker já se **encontrarão instalados**, pelo que não necessita de proceder à sua instalação. Terá apenas de adicionar a imagem do router.

## 2 Instalação do GNS3

Para instalar o GNS3 terá de se registar (é gratuito) e fazer o *download* da versão adequada ao seu sistema operativo a partir de <https://www.gns3.com/>. A versão actual do GNS3 (v.2.2.29) tem suporte para Windows, Linux e Mac OS X. Alguns dos pacotes de instalação trazem não só o GNS3, mas também algumas ferramentas auxiliares tais como programas de emulação de terminais, o programa *wireshark* ou máquinas virtuais.

Embora para este tutorial não sejam necessárias, instale todas as aplicações sugeridas com exceção da máquina virtual. Neste tutorial iremos apenas usar um servidor local, poupando recursos da máquina *host*. Durante a instalação, sempre que lhe surja a opção, escolha apenas **GNS3 local server** ou **Run only legacy IOS on my computer**. O endereço do servidor deverá ser o **127.0.0.1**.

Antes de instalar leia a documentação de suporte à instalação no seu sistema operativo – esta informação está disponível no mesmo ecrã onde é feito o *download*, ilustrado na figura seguinte.

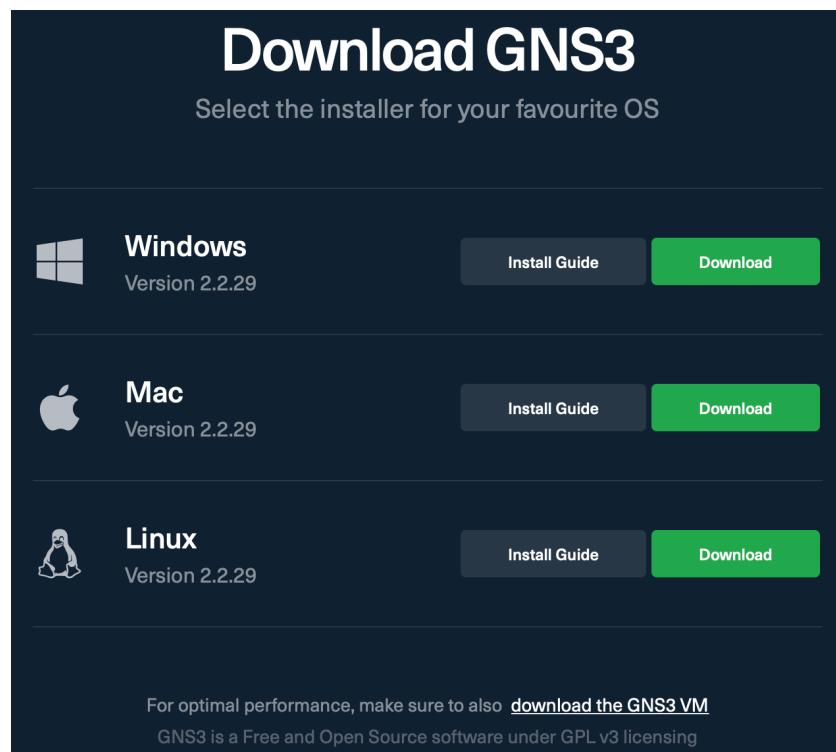


Fig. 1 – Ecrã de download

### Notas:

**1)** Versões mais recentes do Mac OS não trazem o *telnet* que é usado para ligar aos vários dispositivos disponíveis. Para resolver esse problema, o próprio pacote de instalação do GNS3 traz agora uma aplicação de *telnet*. Para saber se tem o *telnet* no sistema abra uma janela de Terminal no Mac OS e escreva: `telnet`. Se o resultado for `telnet: command not found` siga os passos seguintes:

- Opção **GNS3 / Preferences... / General**, aba **Console Applications**, opção **Console Settings**, clicar no botão **Edit**
- Onde está: `telnet %h %p`
- Colocar: `/Applications/GNS3.app/Contents/Resources/telnet %h %p`

**2)** Na versão Windows não necessita de instalar a aplicação SolarWinds.

**3)** Para problemas de instalação consulte a secção 0.

### 3 Instalar imagem do IOS fornecida

Antes de prosseguir faça o *download* do UCStudent da imagem do IOS para *routers* da série Cisco 2600.

#### 3.1 Instalação da imagem

Para instalar esta imagem, aceda às preferências (opção **GNS3 / Preferences...**) através do menu disponível na barra superior do GNS3. Faça em seguida as seguintes modificações:

- Item **Server - Server preferences**, aba **Main Server**, selecionar a opção **Enable Local Server**
- Item **GNS3 VM - GNS3 VM preferences**, não selecionar a opção **Enable the GNS3 VM**
- Item **Dynamips / IOS Routers**
  - Clicar no botão **New**
  - Selecionar a opção **New Image** e escolher o ficheiro com a imagem descarregado do Inforestudante.
  - Escolha a opção **Next** até ao fim da instalação.

A Fig. 2 mostra o ecrã que deverá aparecer depois da instalação da imagem IOS do *router* Cisco2600. Se for necessário clique no símbolo do *router* (seta vermelha na Fig. 2) para que os dispositivos desse tipo apareçam.

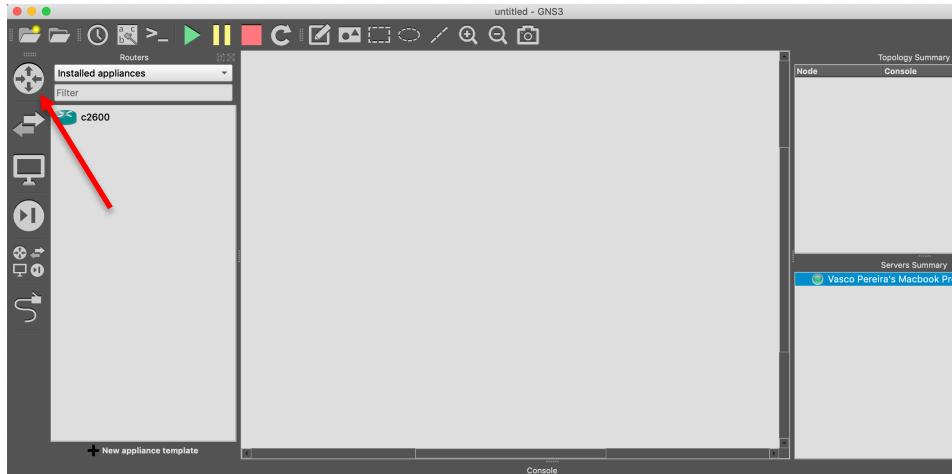


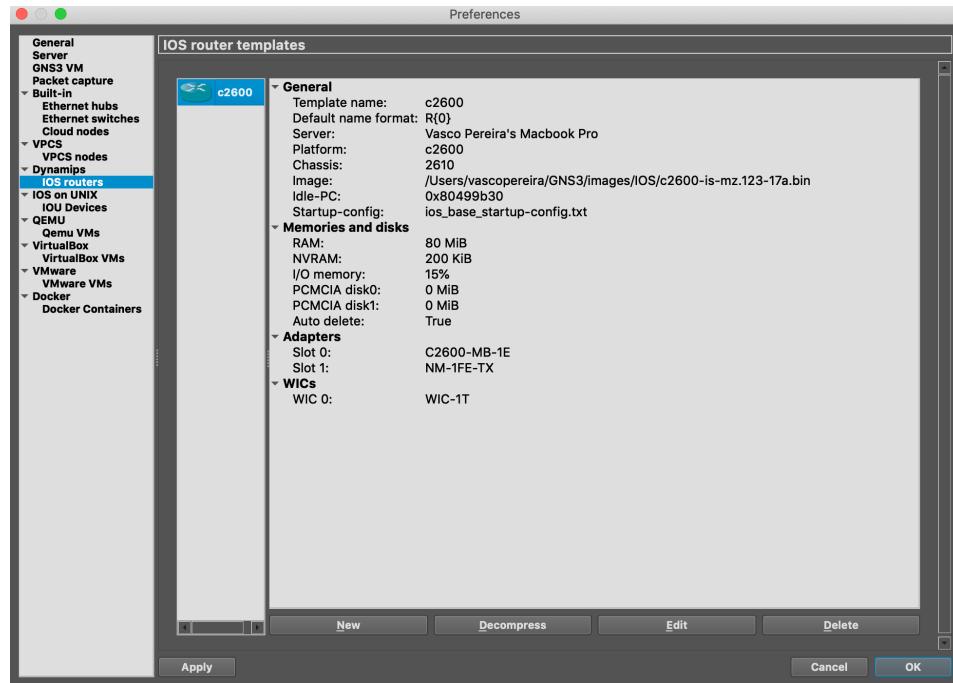
Fig. 2 – Ecrã do GNS3 apόs instalação da imagem do *router* da sérιe Cisco2600.

Depois do *router* instalado clicar sobre ele com o botão direito do rato e escolher a opção **Configure Template**. Altere, caso seja necessário, a configuração da memória, para que consiga correr a imagem disponibilizada no seu computador.

- **Memories and disks:**
  - **RAM size:** 80 MiB ( $1\text{MiB} = 2^{20}$  bytes)
  - **NVRAM size:** 200 KiB (*Non Volatile RAM*) ( $1\text{KiB} = 2^{10}$  bytes)
  - **I/O memory:** 15%
- **Slots:**
  - Inserir no **slot 1** a opção NM-1FE-TX - permite simular uma porta Fast Ethernet (100Mbps)
  - Inserir no **wic 0** a opção WIC-1T - permite simular uma porta serial WAN (Ver Secção 3.2 para detalhes sobre os slots)
- **Advanced:**
  - O **Idle-PC** permite que o emulador não esteja sempre a usar o CPU, forçando a sua paragem durante algum tempo e evitando uma utilização contínua do CPU que reduziria drasticamente a performance da máquina host. Escolha um valor adequado de modo a que o seu CPU nunca esteja a 100%. Poderá especificar um

**Idle-PC** diferente para cada instância do dispositivo que criar na sua topologia. Esta possibilidade será explicada mais à frente.

As configurações alteradas vistas na janela de preferências (**GNS3 / Preferences / IOS routers**) são detalhadas na Fig. 3. Nesta janela, carregando em **Edit**, também poderá aceder à configuração do *router*.



**Fig. 3 – Janela de *Preferences* depois da alteração das configurações da imagem do router Cisco2600.**

Depois de instalada e configurada a imagem, o *router* fica disponível para ser adicionado a qualquer cenário de rede. Ficam disponíveis as seguintes portas:

- Ethernet0/0
- FastEthernet1/0
- Serial0/0

### 3.2 Informação acerca da imagem fornecida

A imagem fornecida no UCStudent é a imagem do IOS para um *router* Cisco da série 2600, Cisco IOS software release 12.3 (17a) com um mínimo de 32 MB DRAM e 8 MB Flash. Este equipamento é ilustrado na figura seguinte.

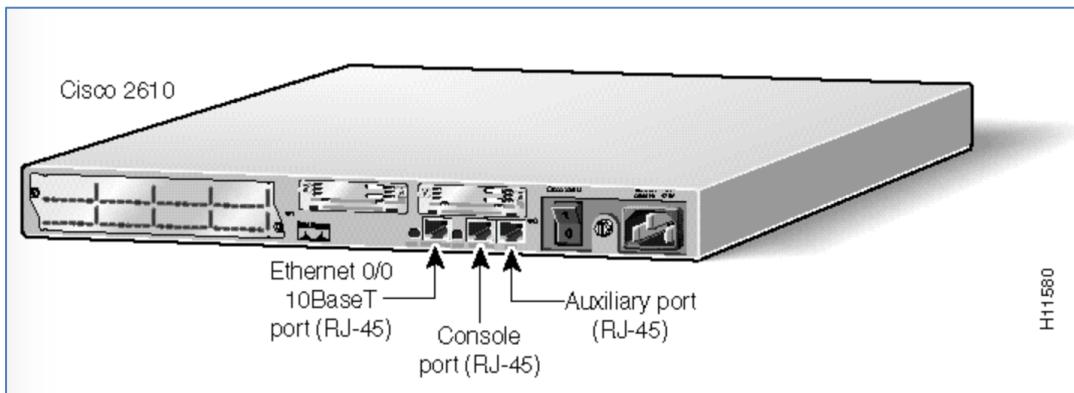


Fig. 4 – Router Cisco 2610

Este *router* tem interfaces por *default* e outras que podem ser adicionadas. Para alterar a configuração do *router* ir a:

Menu **Preferences / Dynamips / IOS routers**; escolher c2600 previamente instalado e clicar no botão **Edit**; mudar as interfaces na opção **slots**.

Como se pode verificar, existe 1 *slot* livre para adaptadores e 3 WICs (*WAN interface Cards*) livres.

#### Adaptadores (Ethernet, FastEthernet):

O CISCO2600 tem por *default* no *slot 0* uma porta *ethernet* que vem sempre no *router* e é usada automaticamente (ver Fig. 4):

- CISCO2600-MB-1E (1 *Ethernet port, automatically used*)

Opções disponíveis para o *slot* adicional:

- NM-1FE-TX (*FastEthernet, 1 port*)
- NM-1E (*Ethernet, 1 port*)
- NM-4E (*Ethernet, 4 ports*)
- NM-16ESW (*Ethernet switch module, 16 ports*)

Outras designações que aparecem noutras *routers* da mesma série:

- CISCO2600-MB-1FE (1 *FastEthernet port, automatically used*)
- CISCO2600-MB-2FE (2 *FastEthernet ports, automatically used*)

Opções disponíveis para o *slot* adicional em outros *routers*:

- NM-4T (*Serial, 4 ports*)

#### WICs (WAN Interface Cards):

Opções disponíveis para os WICs neste router:

- WIC-1T (1 *Serial port*)
- WIC-2T (2 *Serial ports*)

## 4 GNS3

Nesta secção serão descritos o ecrã e os ficheiros usados pelo GNS3.

### 4.1 Ecrã

O ambiente de trabalho do GNS3 divide-se em 7 partes, apresentadas na próxima figura.

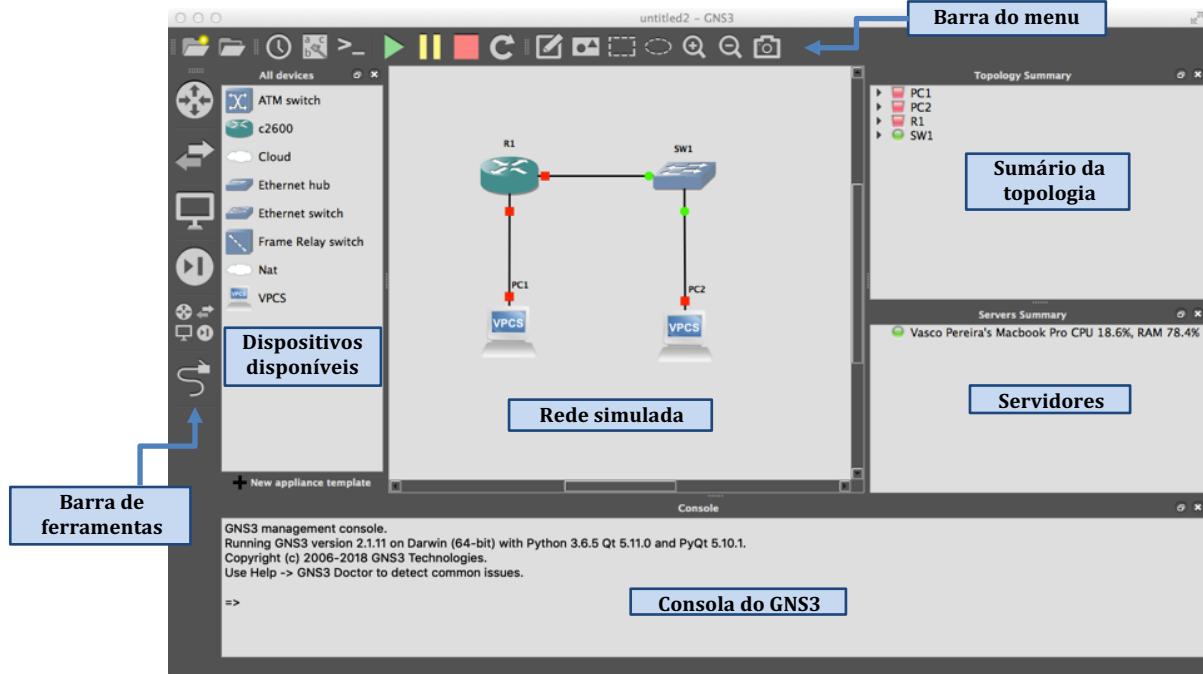


Fig. 5 – Ecrã do GNS3

### 4.2 Ficheiros

Cada projeto é gravado como uma pasta. O nome do projeto pode ser definido pelo utilizador (ex.: `.../GNS3/MyProject`)

Cada pasta de projeto contém um ficheiro com a topologia da rede do projeto `<nome_projeto>.gns3` e outros ficheiros com a configurações dos dispositivos utilizados (*routers* e outros).

### 4.3 Gravação de um projecto

O GNS3 vai automaticamente gravando a topologia e as configurações base dos dispositivos. Alterações efetuadas nos dispositivos (*routers Cisco*) que não sejam gravadas explicitamente na configuração do IOS desse *router* não alteram os ficheiros do projeto, nem são exportadas pela opção **Export config** de cada dispositivo. Para mais detalhes ver Secção 0.

## 5 Uso do Wireshark no GNS3

O Wireshark (<http://www.wireshark.org>) é um analisador de pacotes que permite capturar e analisar comunicações e protocolos, sendo atualmente a aplicação mais popular mundialmente neste contexto. Este *software* é gratuito e já vem incluído em algumas das instalações do GNS3. Pode também instalá-lo manualmente fazendo o download a partir de <http://www.wireshark.org>.

De seguida iremos abordar de que forma o Wireshark será utilizado para capturar e analisar comunicações nos cenários de rede criados com recurso ao GNS3.

### 5.1 Captura de pacotes no GNS3

Para capturar pacotes no GNS3 clique com o botão direito do rato sobre a ligação entre dois dispositivos e selecione **Start capture**. Para terminar a captura escolha a opção **Stop capture**. A figura seguinte ilustra a ativação da captura de pacotes num *link* criado entre dois *routers*, sendo que a opção **Start the capture visualization program** permite arrancar automaticamente o Wireshark, por forma a visualizar os pacotes à medida que eles atravessam o *link* de comunicações, tal como ilustra a Fig. 6. Opcionalmente poderá ser utilizado o Wireshark posteriormente, abrindo o ficheiro com a captura para análise. Para detalhes sobre o funcionamento do Wireshark consulte o manual do *software*.

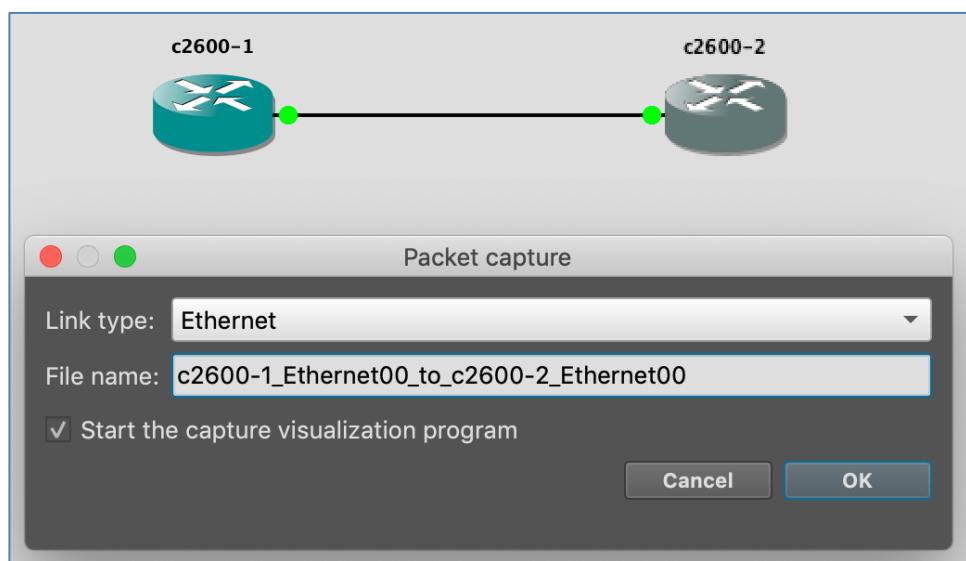
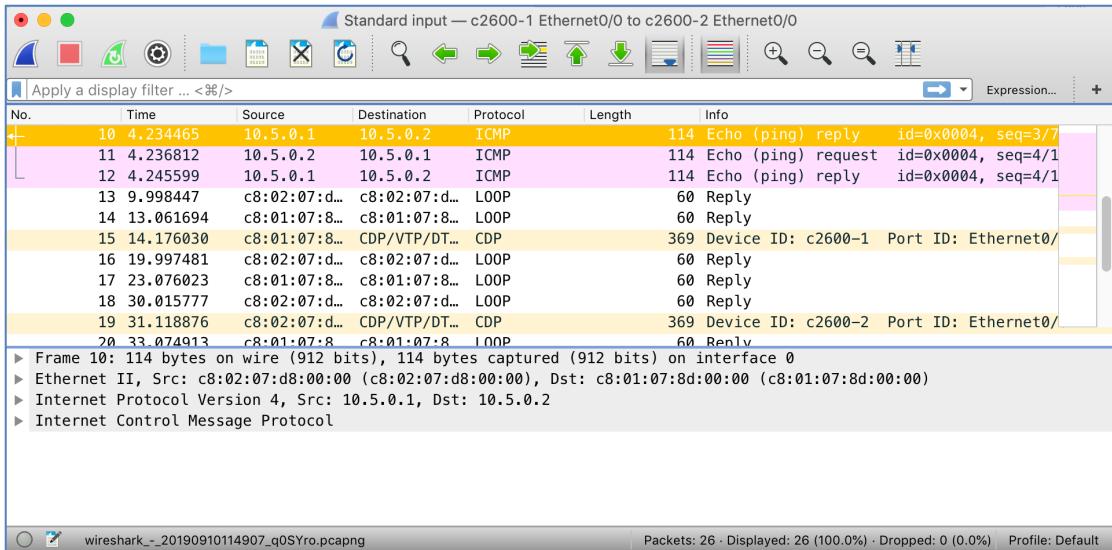


Fig. 6 – Início de captura das comunicações no link entre dois *routers*



**Fig. 7 – Exemplo de captura das comunicações no link criado entre os dois routers**

**Nota:**

- Só conseguirá visualizar tráfego no Wireshark depois de ter configurado pelo menos um dos routers da Fig. 6. A configuração será abordada mais à frente neste documento.

## 6 Comandos básicos a usar nos dispositivos

Nesta secção são descritos alguns dos comandos usados para configurar dispositivos que correm IOS e também os usados para configurar dispositivos VPCS (Virtual PC Simulator).

### 6.1 Dispositivos IOS

Apenas serão abordados uma pequena parte dos comandos disponíveis no IOS. Para comandos adicionais deverá ser consultada bibliografia específica do Cisco IOS.

O IOS tem dois modos de operação básico: modo de utilizador (*user mode*, ou *user exec mode*) e modo privilegiado. O *user mode* permite acesso a informação de estado do *router* enquanto o modo privilegiado dá acesso a toda a informação do *router*, incluindo os comandos e ficheiros de configuração. Cada modo apresenta uma *prompt* específica:

```
Router>      ; user mode
Router#     ; modo privilegiado
```

A passagem ao modo privilegiado faz-se através do comando **enable** e mediante uma *password* (se necessário).

```
Router>enable
Password:
Router#
```

Para alterar a configuração de um dispositivo IOS tem que se entrar no sub-modo de configuração, através do comando *configure terminal*:

```
Router#configure terminal
Enter configuration commands, one per line. End with Ctrl-Z
Router(config) #
```

Este modo de configuração tem ainda vários sub-modos como o que permite configurar as diversas interfaces - comando **interface**.

Os comandos disponíveis em qualquer modo podem ser visualizados digitando “?”.

De seguida é exemplificada a configuração da interface ethernet0/0 de um *router* com a designação “Router0”.

```
Router0>enable          // para entrar no modo privilegiado
Router0#show running-config // para ver as interfaces presentes no router
                           // use a barra de espaço para mudar a
                           // página de configuração
(. . .)
Router0#configure terminal           // para entrar em modo de configuração
Router0(config)#interface Ethernet0/0 // configurar interface Ethernet0/0
Router0(config-if)#ip address 10.254.0.2 255.255.255.0
Router0(config-if)#no shutdown       // activar a configuração da interface
Router0(config-if)#exit
Router0(config)#exit
```

O comando **ip address** define o endereço IP de uma interface:

```
ip address ip-address mask
```

É importante perceber que os *routers* reconhecem, de forma automática, as redes às quais estão diretamente ligados, i.e., as redes ligadas às suas várias interfaces. Isto significa que conseguem comunicar diretamente com outros sistemas ou *routers* ligados a essas mesmas redes, bastando para tal que todos os equipamentos estejam corretamente configurados com um endereço IP (único na rede) e com a máscara de rede comum à rede.

Por outro lado, para encaminharem pacotes para outras redes diferentes, têm de ser definidas rotas, que permitam ao *router* saber para onde deve enviar os pacotes, no seu caminho até ao

sistema de destino. As rotas podem ser **estáticas** (configuradas manualmente nos equipamentos) ou **dinâmicas** (configuradas por protocolos de encaminhamento ou de *routing*).

Para definir **rotas estáticas** (neste momento não iremos abordar protocolos de *routing*) iremos usar o comando `ip route`, cuja sintaxe é a seguinte:

```
ip route destination_network subnet_mask default_gateway
```

Para remover uma rota, use em modo de configuração o seguinte comando:

```
no ip route destination_network subnet_mask default_gateway
```

A configuração de encaminhamento do *router* pode ser visualizada recorrendo ao comando:

```
Router0#show ip route
```

Os routers Cisco têm duas configurações distintas. A **running configuration** é a configuração que está a ser executada e reflete todos os comandos que foram usados. Como se encontra armazenada na memória RAM do dispositivo é perdida assim que ele é desligado. A **startup configuration** é a configuração de arranque, guardada na memória não volátil do *router*. É com esta configuração que o *router* começa a executar no arranque.

Para copiar a **running configuration** existente no *router* para a **startup configuration** execute o seguinte comando:

```
Router0# copy running-config startup-config
```

#### Dicas de utilização do IOS:

- Para repor a configuração *default* de uma interface (ex: Ethernet0/0), durante a configuração fazer: `default interface Ethernet0/0`
- A tecla 'Tab' completa os comandos
- Vários comandos podem ser abreviados
  - Ex:  
`conf t` (ou outras variações) em vez de `configure terminal`  
`sh runn` em vez de `show running-config`
- Clique na seta para cima para aceder aos comandos anteriores
- Carregue na tecla 'q' para sair de um *scroll* paginado

## 6.2 Virtual PC Simulator (VPCS)

Uma maneira de simular um computador (*host*) no GNS3 é usando dispositivos VPCS. Embora estes dispositivos não simulem completamente um computador, para isso será necessário ligar máquinas reais ou usar máquinas virtuais no GNS3, permitem as funcionalidades básicas de rede que permitem testar a conectividade das redes criadas no GNS3. Para utilizar dispositivos do tipo VPCS no GNS3 bastará clicar nos **End Devices** e arrastar o **VPCS** para o cenário, tal como ilustra a figura seguinte. Para mais informação consultar <https://docs.gns3.com>.

**Nota:** os dispositivos VPCS não são instalados automaticamente durante a instalação do GNS3 no Linux, pelo que os terá de instalar posteriormente.

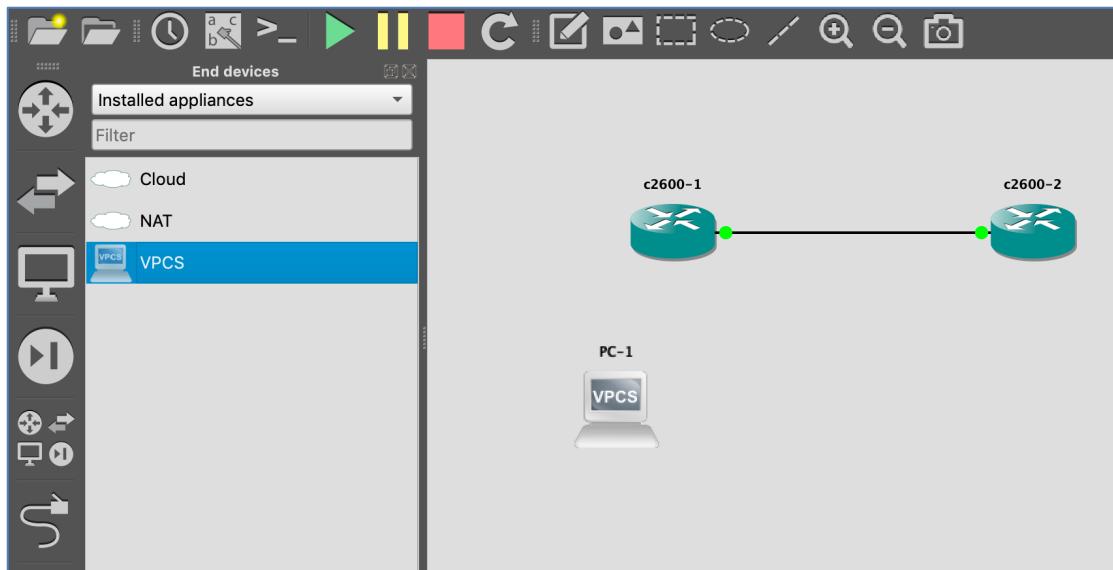


Fig. 8 – Adição de um dispositivo do tipo VPCS ao cenário no GNS3

Alguns comandos que podem ser usados na consola da VPCS:

? – lista todos os comandos possíveis

comando ? – ajuda sobre o comando

show – mostra a configuração do host

Ex:

```
PC1> show ip  
PC1> show ip all
```

ip {end.IP} {máscara rede} {gateway} – define o endereço ip, máscara e gateway do host

Ex: PC1> ip 192.168.1.10 255.255.255.0 192.168.1.254

ip {endereço IP usando CIDR} {gateway} – define o endereço IP, máscara e gateway do host

Ex: PC1> ip 192.168.1.10/24 192.168.1.254

dhcp – configura um host para usar DHCP (configuração dinâmica da rede)

save – guarda a configuração do VPCS na pasta do projecto (se não for gravado é perdido da próxima vez que se fechar o projecto)

quit – fecha o VPCS

ping {endereço IP} - testar conectividade ao endereço especificado

Outros comandos que podem ser usados: arp, trace

## 7 Primeira simulação de uma rede no GNS3

Nesta secção vamos construir uma rede de topologia simples e configurar todos os dispositivos que representam equipamento ativo de rede. Os *routers* a utilizar serão baseados no c2600 configurado na Secção 3. Essa rede e todo o equipamento ativo serão depois simulados.

### 7.1 Cenário a simular

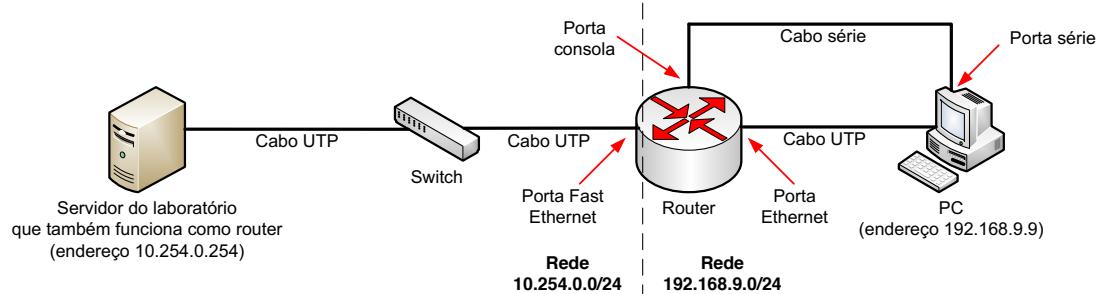


Fig. 9 – Cenário a simular

O cenário a simular encontra-se ilustrado na Fig. 9 e está dividido em 2 redes. Vamos designar uma como **rede do laboratório** – aquela onde se encontra o Servidor do laboratório e que tem o endereço 10.254.0.0/24, e outra como **rede local** – aquela onde se encontra o PC e que tem o endereço 192.168.9.0/24.

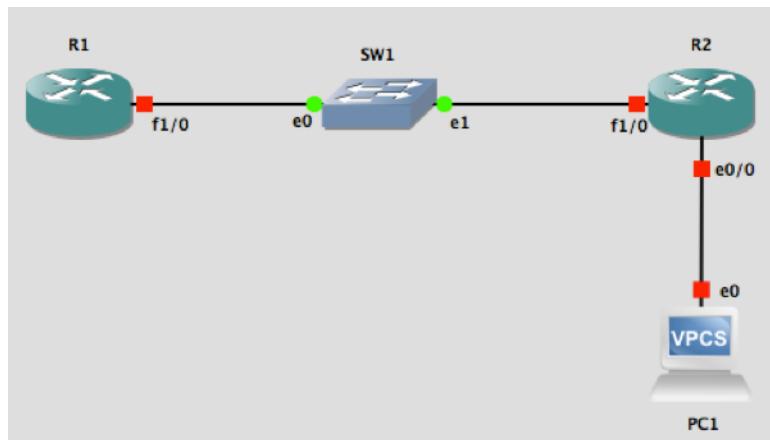
### 7.2 Construção da rede

Para simular o cenário proposto vamos usar 4 dispositivos:

- 1 *router* c2600 – vai emular o *router* local
- 1 *router* c2600 – vai emular o *router* do laboratório; como neste tutorial apenas se pretendem emular as capacidades de *routing* do servidor do laboratório, o *router* chegará para simular a totalidade do servidor
- 1 *switch ethernet* – vai emular o *switch*
- 1 VPCS (Virtual PC Simulator) – vai simular a ligação através de um cabo *ethernet* entre o PC e o *router* local. Para a configuração do *router* usaremos a própria consola do dispositivo emulado no GNS3 - permitirá a configuração através de um cabo série ligado entre a porta série do PC e a porta da consola do *router*.

Para fazer a ligação entre os vários dispositivos serão usados *links* que representam canais físicos de comunicação.

Comece por escolher os vários dispositivos necessários na barra de ferramentas presente na barra esquerda e arraste-os para o retângulo central (a área com o cenário da rede simulada). De modo a interligar os vários dispositivos usa a ferramenta **Add a Link** disponível na barra à esquerda do GNS3, escolhendo para cada extremo a interface correta de modo a obter a rede da Fig. 10.



**Fig. 10 – Cenário a simular com ligação entre os vários interfaces**

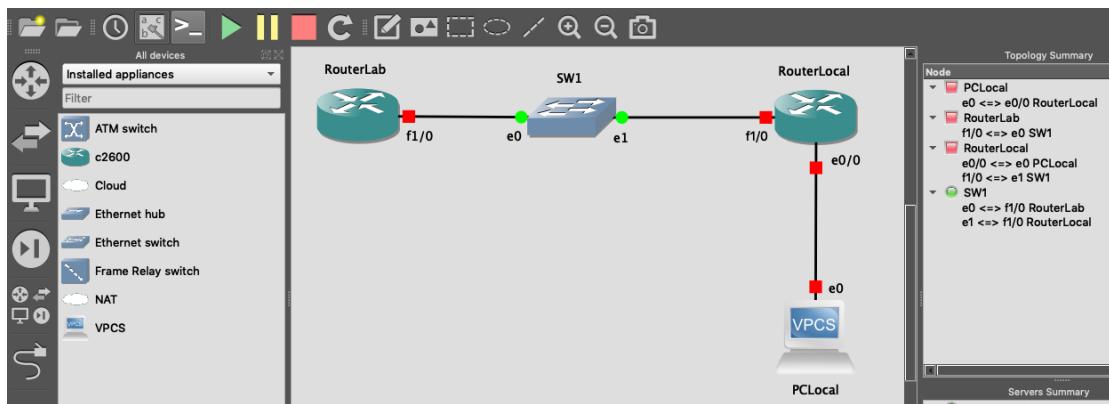
**Notas:**

- Para visualizar as interfaces aceda ao menu **View / Show interface labels**
- Pode mover as etiquetas com a descrição de cada interface/dispositivo, bastando para tal arrastá-las.

### 7.3 Configurações

Com os dispositivos todos desligados efetue os seguintes passos:

- Clique com o botão direito do rato em cima de cada um dos dispositivos, aceda à opção **Configure** ou **Change hostname** e mude-lhes o nome, de modo a ficarem como na Fig. 11 (deste modo será mais fácil identificar os dispositivos – este nome também aparecerá na barra da janela da consola quando abrir a consola do dispositivo).



**Fig. 11 – GNS3 com dispositivos e ligações**

De seguida, vamos ligar todos os dispositivos (corresponde a iniciar o seu funcionamento). Pode ligar um a um usando a opção **Start** existente no menu de contexto de cada dispositivo (acessível através do botão direito do rato) ou podem-se ligar todos de uma vez acedendo ao menu **Control / Start / Resume all nodes**.

Ao ligar os dispositivos, ficará visível uma bola verde do lado esquerdo do seu nome, na área **Topology Summary** do ecrã, bem como no símbolo das interfaces no retângulo central (ver Fig. 11, que ilustra um cenário onde só o switch está ligado).

Podemos, nesta fase, especificar também os valores do **Idle-PC** para cada dispositivo. Tal só deverá ser necessário caso verifique que o GNS3 está a exigir demasiado poder de processamento ao computador. Para tal, aceda ao menu de contexto e use a opção **Idle-PC** para definir diretamente os valores ou, em alternativa, ative a opção **Auto Idle-PC** para obter um valor automaticamente. Tenha em atenção que nem todos os dispositivos têm esta funcionalidade.

Para configurar os dispositivos vamos agora abrir consolas específicas para cada um. Para isso, aceda ao menu **Control / Console connect to all nodes**.

Antes de proceder à configuração de cada um dos dispositivos aceda à consola do PCLocal e execute o seguinte comando:

```
VPCS> ping 10.254.0.254
```

A resposta obtida será:

```
host (10.254.0.254) not reachable
```

Verifica-se que neste momento não temos conexão ao endereço 10.254.0.254 (Servidor e *router* do laboratório). Vamos então proceder às configurações necessárias, seguindo os passos que se exemplificam a seguir.

### Configuração do router do laboratório

- O *router* do laboratório deverá ter a interface que liga à rede do laboratório (10.254.0.0/24) com endereço igual a 10.254.0.254 (ver Fig. 9).
- Execute na consola do *router* do laboratório:

```
RouterLab>enable
RouterLab#show running-config (para ver os nomes das interfaces presentes no router)
RouterLab#config terminal
RouterLab(config)#interface FastEthernet1/0
RouterLab(config-if)#ip address 10.254.0.254 255.255.255.0
RouterLab(config-if)#no shutdown
RouterLab(config-if)#exit
RouterLab(config)#exit
```

### Configuração do router local

- O *router* local terá na interface para a rede do laboratório um endereço qualquer da gama disponível na rede 10.254.0.0/24. Vamos atribuir-lhe por exemplo o endereço 10.254.0.9. O endereço da interface para a rede local (onde apenas está o PC) poderá ser o 192.168.9.254, pertencente à rede 192.168.9.0/24.
- Antes de avançar, de modo a ter a certeza de qual foi a interface escolhida para a rede local e para a rede do laboratório, vamos observar quais as ligações estabelecidas na área da *Topology Summary*, conforme ilustrado na Fig. 12.

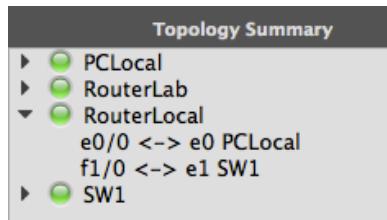


Fig. 12 - Sumário da topologia

Podemos ver que a interface para a rede do laboratório é o f1/0 e para a rede local é o e0/0. Podemos então iniciar agora a configuração.

Comecemos pela interface para a rede do laboratório, utilizando a consola para o *router* local:

```
RouterLocal>enable
RouterLocal#show running-config (para ver as interfaces presentes no router)
RouterLocal#config terminal
RouterLocal(config)#interface FastEthernet1/0
RouterLocal(config-if)#ip address 10.254.0.9 255.255.255.0
RouterLocal(config-if)#no shutdown
RouterLocal(config-if)#exit
RouterLocal(config)#exit
```

Para verificar que já se consegue aceder ao *router* do laboratório a partir do *router* local podemos usar o comando **ping**.

Execute na consola do *router* local:

```
RouterLocal> ping 10.254.0.254
```

O resultado deverá ser algo parecido com:

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.254.0.254, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 64/72/92 ms
```

Podemos fazer também **ping** do *router* do laboratório para o local. Execute na consola do *router* do laboratório:

```
RouterLab> ping 10.254.0.9
```

Vamos agora configurar a parte da rede local no *router* local. Execute na consola do *router* local:

```
RouterLocal>enable
RouterLocal#show running-config (para ver as interfaces presentes no router)
RouterLocal#config terminal
RouterLocal(config)#interface Ethernet0/0
RouterLocal(config-if)#ip address 192.168.9.254 255.255.255.0
RouterLocal(config-if)#no shutdown
RouterLocal(config-if)#exit
RouterLocal(config)#exit
```

Antes de prosseguir verifique no *router* local o estado das interfaces.

```
RouterLocal#show running-config
```

Deverá obter os seguintes dados (entre outros):

```
(...)
interface Ethernet0/0
  ip address 192.168.9.254 255.255.255.0
  half-duplex
!
interface Serial0/0
  no ip address
  shutdown
!
interface FastEthernet1/0
  ip address 10.254.0.9 255.255.255.0
  duplex auto
  speed auto
!
(...)
```

Falta agora configurar o PC para que consigamos fazer **ping** ao *router* do laboratório. Para isso precisamos de lhe dar um endereço, uma rede e um *default gateway*.

**Nota:** o *default gateway* é o endereço para onde serão enviados os pacotes de rede cujo IP pertence a uma rede que o PC não conhece.

A rede será a rede local, já definida anteriormente como 192.168.9.0/24, o *gateway* será o endereço do *router* local (192.168.9.254) e o PC terá de ser qualquer endereço dentro da gama da rede local – na Fig. 9 foi definido o 192.168.9.9.

Execute na consola da VPCS (PCLocal):

```
VPCS> ip 192.168.9.9/24 192.168.9.254
```

O resultado será:

```
Checking for duplicate address...
PC1 : 192.168.9.9 255.255.255.0 gateway 192.168.9.254
```

Para verificar a configuração da VPCS execute na consola da VPCS:

```
VPCS> show ip
```

O resultado obtido será semelhante a:

NAME	:	VPCS[1]
IP/MASK	:	192.168.9.9/24
GATEWAY	:	192.168.9.254
DNS	:	
MAC	:	11:11:11:11:11:11
LPORT	:	10004
RHOST:PORT	:	127.0.0.1:10005

MTU: : 1500

Neste momento temos todas as interfaces configuradas. No entanto, ainda não conseguiremos fazer **ping** desde o PC até ao *router* do laboratório, visto que o *router* do Laboratório não tem conhecimento da rede 192.168.9.0/24. Cada *router* só conhece as redes a que está diretamente ligado.

Para resolver este problema vamos acrescentar uma rota estática ao *router* do laboratório. Esta rota permitirá a este *router* encaminhar os pacotes que têm como destino a rede 192.168.9.0/24 (a que o *router* do laboratório não está diretamente ligado e que por isso não conhece) através do *router* local. Para criar rotas estáticas vamos usar o comando `ip route`.

```
ip route destination_network subnet_mask default_gateway
```

No *router* do laboratório execute:

```
RouterLab>enable
RouterLab#config terminal
RouterLab(config)#ip route 192.168.9.0 255.255.255.0 10.254.0.9
RouterLab(config-router)#exit
```

Para verificar se as configurações estão correctas execute o seguinte comando:

```
RouterLab#show ip route
```

O resultado será:

```
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
      D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
      N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
      E1 - OSPF external type 1, E2 - OSPF external type 2
      i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS
      level-2
      ia - IS-IS inter area, * - candidate default, U - per-user static
      route
      o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

S      192.168.9.0/24 [1/0] via 10.254.0.9
      10.0.0.0/24 is subnetted, 1 subnets
C          10.254.0.0 is directly connected, Ethernet0/0
```

Poderá ver, no resultado anterior, a indicação das rotas ativas no *router*: rotas estáticas assinaladas com “**S**” e rotas para redes diretamente ligadas a interfaces do *router* com “**C**”. Temos neste momento tudo configurado pelo que vamos voltar a testar o **ping** inicial feito desde o PC até ao *router* do laboratório. Execute na consola da VPCS:

```
VPCS> ping 10.254.0.254
```

A resposta obtida será parecida com:

```
84 bytes from 10.254.0.254 icmp_seq=1 ttl=254 time=69.273 ms
84 bytes from 10.254.0.254 icmp_seq=2 ttl=254 time=20.854 ms
84 bytes from 10.254.0.254 icmp_seq=3 ttl=254 time=18.616 ms
84 bytes from 10.254.0.254 icmp_seq=4 ttl=254 time=15.825 ms
84 bytes from 10.254.0.254 icmp_seq=5 ttl=254 time=16.458 ms
```

Teste também o comando **ping** do *router* do laboratório até ao PC. Execute na consola do *router* do laboratório:

```
RouterLab>ping 192.168.9.9
```

A resposta obtida será parecida com:

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.9.9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 16/20/28 ms
```

Neste tutorial não foi necessária qualquer configuração do *switch*. A topologia final da rede com os endereços dos vários dispositivos é apresentada na Fig. 13 (os endereços IP foram colocados na imagem recorrendo a *Notes*: menu **Annotate / Add Note**).

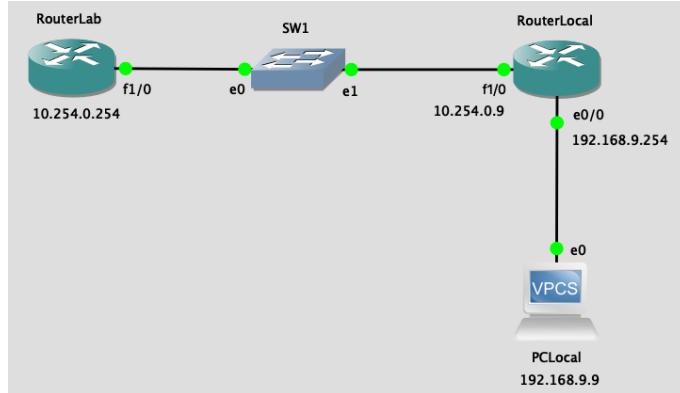


Fig. 13 – Topologia e endereços

## 7.4 Captura de pacotes usando o Wireshark

**Nota:** Antes de iniciar esta parte certifique-se que tem o *Wireshark* instalado.

Clique com o botão direito sobre a ligação entre o *router* do laboratório e o *switch* e escolha a opção **Start capture**. Ao escolher esta opção o wireshark será automaticamente aberto e começará a analisar todos os pacotes que passam nesse troço de rede (*live capture*). Para ver algum tráfego execute o seguinte comando no *router* do laboratório:

```
RouterLab>ping 192.168.9.9
```

A resposta obtida na consola do *router* do laboratório será parecida com:

```
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.9.9, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 64/90/128 ms
```

Na janela do *Wireshark* deverá obter um resultado semelhante ao ilustrado na Fig. 14:

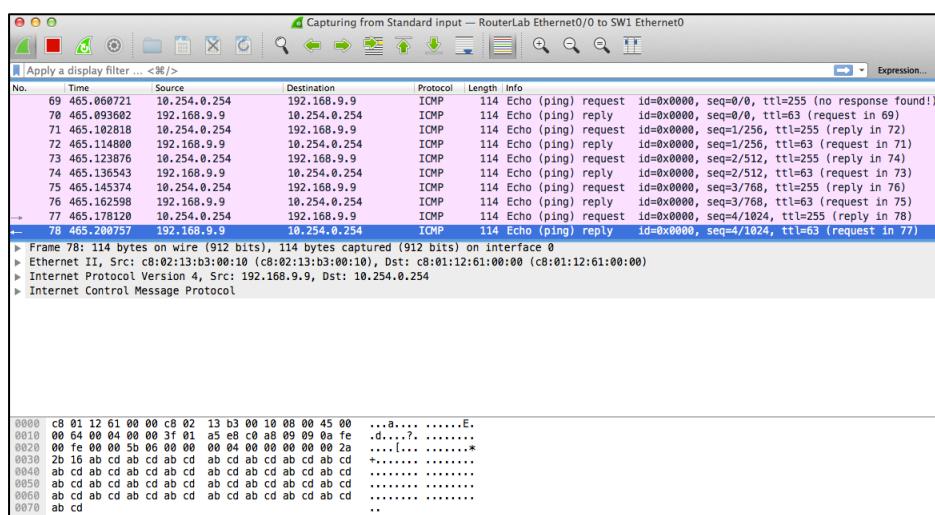


Fig. 14 - Janela do Wireshark

## 8 Containers no GNS3

O uso de *containers* aumenta as possibilidades de simulação do GNS3, possibilitando a interligação de máquinas dentro da rede virtualizada. Esta secção detalha a instalação do Docker numa máquina Linux, mais concretamente num sistema LUbuntu.

### 8.1 Instalação do Docker – exemplo do Linux (LUbuntu)

Entre no Linux usando o seu utilizador, abra uma consola e digite o seguinte comando:

```
$ sudo apt install docker.io
```

Este comando irá instalar a versão mais recente do Docker existente nos repositórios.

Ser-lhe-á pedida a password de root e de seguida terá de responder com "Y" à pergunta feita no script de instalação. De seguida escreva o seguinte comando para adicionar o seu utilizador ao grupo *docker*:

```
$ sudo adduser {login do seu user} docker
```

Para concluir a instalação reinicialize a máquina através do comando:

```
$ reboot
```

Após a máquina reiniciar já terá o *Docker* instalado no sistema.

### 8.2 Criação de um container

Para criar um *container* precisamos de definir primeiro um ficheiro Dockerfile, o qual vai definir o que iremos colocar no *container*. Como exemplo, iremos criar um *container* com uma imagem base do Ubuntu 20.04 e que terá instaladas algumas ferramentas.

Crie um ficheiro de texto com o nome Dockerfile.

Escreva as seguintes linhas:

```
FROM ubuntu:20.04
RUN apt-get update && apt-get install -y nmap iputils-ping gcc gdb make
net-tools traceroute netcat
```

Para conseguir partilhar uma pasta entre o *container* e o sistema *host*, acrescente ainda (mais informação será fornecida na secção 8.3):

```
RUN chmod 777 /home
WORKDIR /gns3volumes/home
```

(para saber mais sobre os comandos possíveis num Dockerfile aceda à documentação do Docker em <https://docs.docker.com>)

Para criar o *container* entre na mesma pasta onde está o "Dockerfile". De seguida faça o *build* usando o comando

```
docker build -t {nome do container} {pasta onde está o Dockerfile}
```

Exemplo:

```
$ docker build -t dei/rc_container .
```

#### Alguns comandos do Docker:

docker --version	- mostrar a versão instalada do Docker
docker run -it -d {image name}	- criar um <i>container</i> a partir de uma imagem
docker ps -a	- mostrar todos os <i>containers</i>
docker stop {container id}	- parar a execução de um <i>container</i>

<code>docker kill {container id}</code>	- terminar a execução de um <i>container</i> imediatamente
<code>docker images</code>	- listar todas as Docker <i>images</i> existentes na máquina
<code>docker rm {container id}</code>	- remover um <i>containers</i>
<code>docker rmi {image id}</code>	- remover uma imagem da <i>storage</i> local
<code>docker build {path to dockerfile}</code>	- fazer o <i>build</i> de uma imagem a partir do Dockerfile

### 8.3 Instalação no GNS3 do *container* criado

Para instalar o *container* criado na secção anterior ao GNS3, aceda às preferências (opção **GNS3 / Preferences...**) através do menu disponível na barra superior do GNS3. Faça em seguida as seguintes modificações:

- Item **Docker / Docker containers**
  - Clicar no botão **New**
  - Estando selecionado o **Existing image** selecionar o *container* criado anteriormente (neste exemplo o `dei-rc_container`).
  - Escolha a opção **Next**. Nos passos seguintes poderá configurar o *container* no GNS3. Pode escolher os valores por omissão escolhendo sempre a opção **Next**.
    - Opção **Name**: permite alterar o nome do *container* no GNS3;
    - Opção **Adapters**: permite escolher o número de interfaces de rede do *container*

O novo *container* pode ser visto na janela de preferências (**GNS3 / Preferences / Docker / Docker containers**) - Fig. 15. Nesta janela, carregando em **Edit**, poderá aceder à configuração actual do *container*.

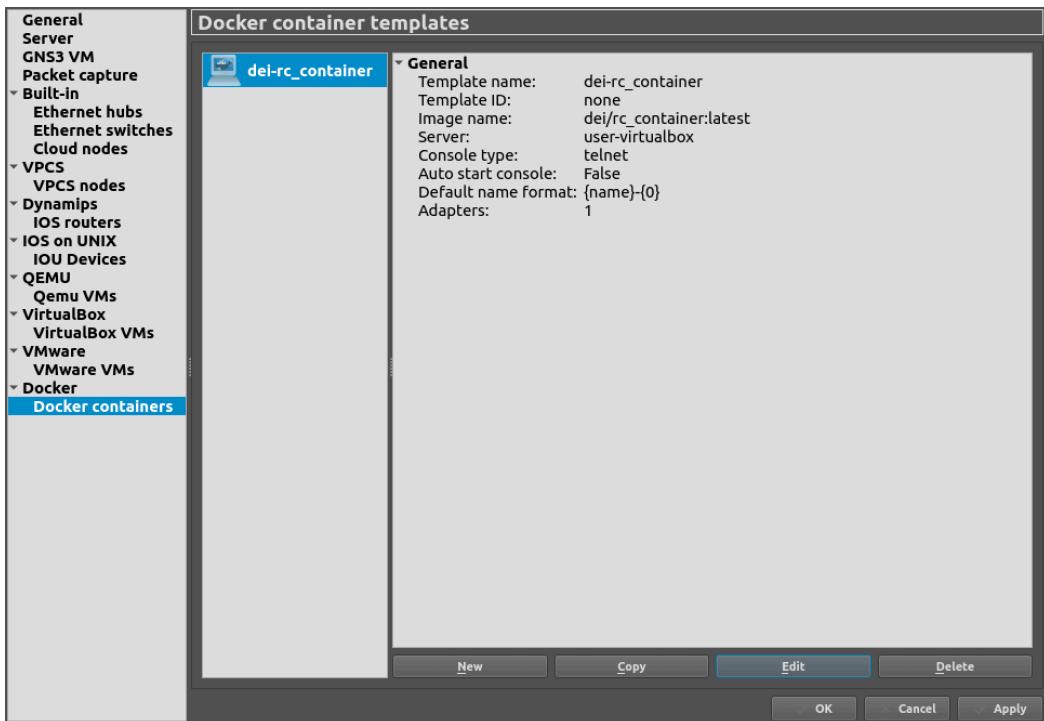


Fig. 15 - Janela de **Preferences** depois de criado o novo *container*.

Para mapear uma pasta do sistema *host* no *container* criado no GNS3, vamos agora alterar algumas das configurações por omissão. Na janela da Fig. 15 clique no botão **Edit**, na aba **Advanced** e, na opção **Additional directories to make persistente [...]**, insira a pasta com o nome `/home`.

A nova configuração será agora a detalhada na Fig. 16.

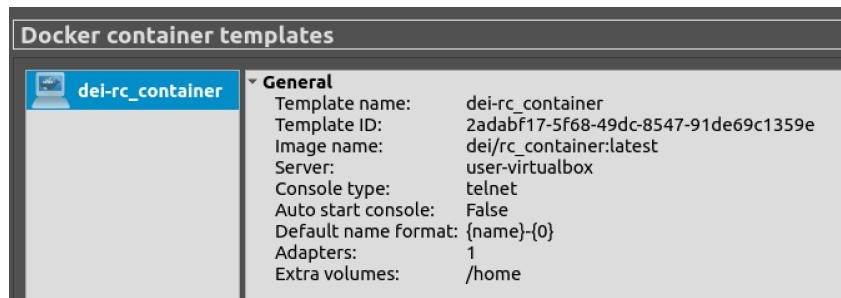


Fig. 16 – Nova configuração do *container* *dei-rc\_container*

Ao criar um projeto no GNS3 que contenha este *container*, na pasta respetiva desse projeto localizada no *host*, para cada instância do *container* irá agora aparecer a pasta *home*. Poderá através dessa pasta transferir ficheiros do *host* para o *container* e vice-versa. Esta pasta também é diretamente acessível clicando com o botão direito do rato no *container* existente na simulação e escolhendo a opção **Show in file manager**.

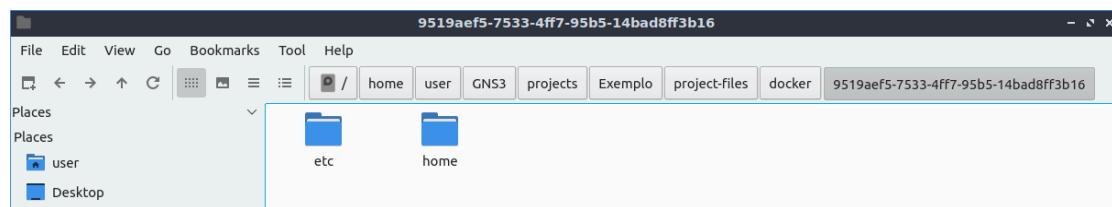


Fig. 17 – Pasta do *container* instanciado no GNS3, acedida através do sistema *host*

A pasta será também visível dentro do *container* accedendo através da consola à pasta */gns3volumes/home* (Fig. 18).



Fig. 18 – Acesso ao *container* a partir do GNS3, usando a consola

## 9 Problemas de instalação

- Mensagem “Waiting for http://127.0.0.1 (...)” aparece em ciclo.

### Solução:

Verificar se o GNS3 é permitido pelo antivírus e pela firewall. Como alternativa, desligue temporariamente o antivírus e a *firewall* – não se esqueça de reactivar ambos após usar o GNS3.

- Router c2600 não arranca. Ligando pela consola aparece o ecrã da figura seguinte e o router fica bloqueado.

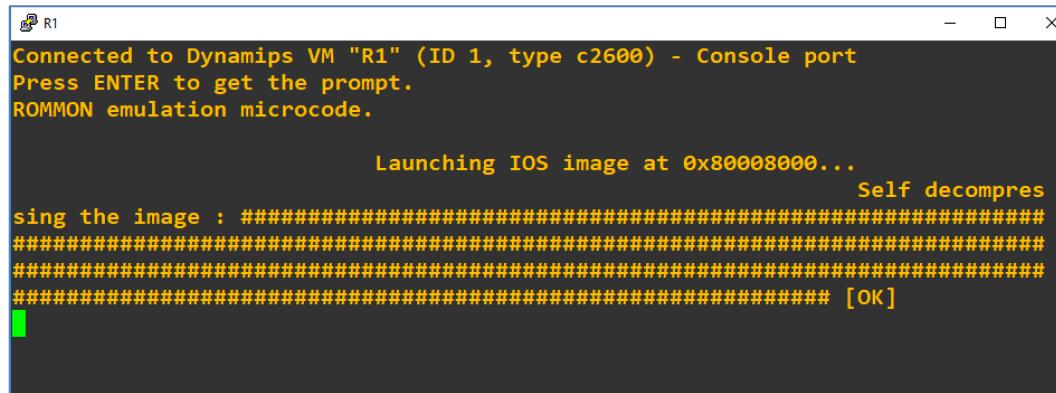


Fig. 19 – Consola ligada ao router R1

### Solução:

Verificar os valores de memória na configuração do router (ver secção 3.1).

- Com sistemas *host* Windows a aplicação de *telnet* instalada por omissão com o GNS3 sai fora do ecrã.

### Solução:

Trocar de programa de *telnet*. No Windows o programa **Putty** também é instalado juntamente com o GNS3. Para o usar deverá alterar o comando predefinido para a aplicação de consola – ver figura seguinte.

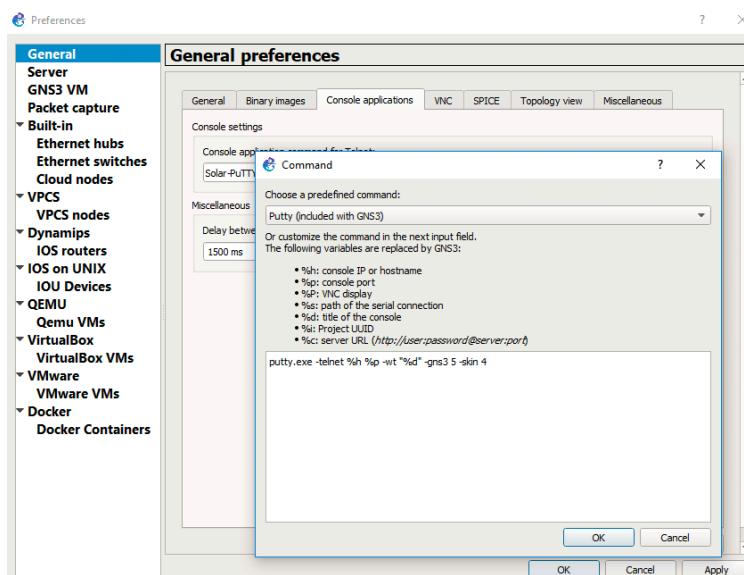


Fig. 20 – Alteração da aplicação de consola para o Putty

- Após a instalação do GNS3 num sistema *host* Windows, aparece um erro a dizer que o Winpcap ou o Npcap, não foram instalados (Fig. 21).

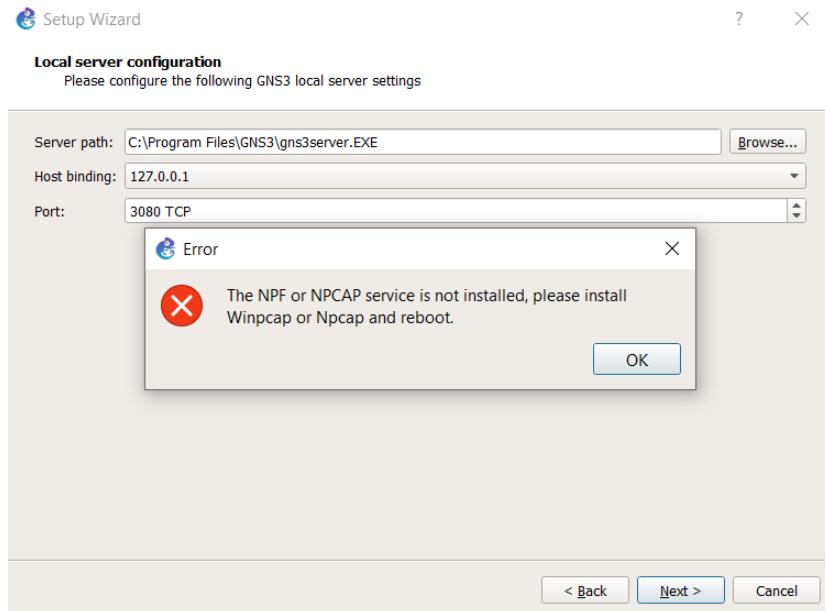


Fig. 21 – Erro no Winpcap ou Npcap

#### Solução:

Certifique-se que durante a instalação do GNS3 no Windows as opções **winpcap** e/ou **npcap** estavam selecionadas.

Caso estejam efetivamente instalados, execute o **cmd.exe** (*command prompt*) como administrador:

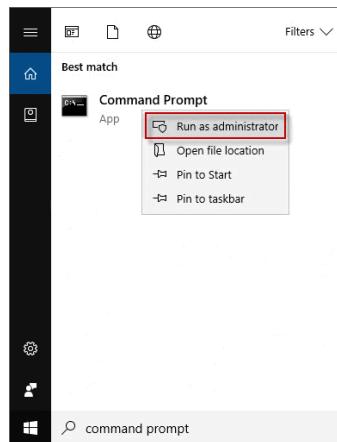


Fig. 22 – Executar a *command prompt* como administrador

De seguida execute os seguintes comandos:

```
net stop npf  
net start npf
```

- Ao correr o Wireshark no Linux dá o erro: "Couldn't run /usr/bin/dumpcap in child process: Permission denied"

#### Solução:

O Wireshark está a informar que precisa de correr como root para aceder às interfaces de rede. Para isso, abrir uma janela de terminal e fazer:

```
$ sudo dpkg-reconfigure wireshark-common
```

Escolher “yes” para dar permissão a qualquer utilizador. De seguida dar permissões de execução ao `dumpcap` escrevendo na janela do terminal:

```
$ sudo chmod +x /usr/bin/dumpcap
```

- O GNS3 está permanentemente a mostrar uma janela que diz o erro: “waiting for `http://127.0.0.1:3080`” ou “waiting for `http://localhost:3080`”

Tente os seguintes passos incrementalmente, i.e., avance para o seguinte apenas se o anterior não resolver o problema:

1 – Dê permissões ao GNS3 na sua firewall

2 --Desligue a firewall

3 – Desligue o antivírus

Nota: Nos casos 2 e 3, por motivos de segurança, desligue primeiro o seu PC da rede; volte a repor as configurações iniciais antes de se ligar novamente à rede.