Sobre a licença

creative commons

Atribuição-Compartilhamento pela mesma Licença 2.5 Brasil

Você pode:

- copiar, distribuir, exibir e executar a obra
- criar obras derivadas

Free Cultural APPROVED FOR Works

Sob as seguintes condições:

- **Atribuição**. Você deve dar crédito ao autor original, da forma especificada pelo autor ou licenciante.
- **Compartilhamento pela mesma Licença**. Se você alterar, transformar, ou criar outra obra com base nesta, você somente poderá distribuir a obra resultante sob uma licença idêntica a esta.

- Para cada novo uso ou distribuição, você deve deixar claro para outros os termos da licença desta obra.
- Qualquer uma destas condições podem ser renunciadas, desde que Você obtenha permissão do autor.
- Nothing in this license impairs or restricts the author's moral rights.

Para cada novo uso ou distribuição, você deve deixar claro para outros os termos da licença desta obra.
No caso de criação de obras derivadas, os logotipos do CGI.br, NIC.br, IPv6.br e CEPTRO.br não devem ser utilizados.
Na atribuição de autoria, essa obra deve ser citada da seguinte forma:
Apostila “Curso IPv6 básico” do NIC.br, disponível no sítio http://curso.ipv6.br ou através do e-mail ipv6@nic.br.
Qualquer uma destas condições podem ser renunciadas, desde que você obtenha permissão do autor.
Se necessário, o NIC.br pode ser consultado através do email ipv6@nic.br.
Nada nesta licença prejudica ou restringe os direitos morais do autor.

Publicação

ceptro.br

IPv6.br

Curso IPv6 básico

Laboratório: Familiarizando-se com o IPv6

cgi.br nic.br

Laboratório – Familiarizando-se com o IPv6

Objetivo: Familiarizarmos com as novas características do protocolo IPv6, configurando-o em nossos notebooks e realizando os primeiros exercícios do laboratório, como configuração manual de endereços e rotas. Também analisaremos a estrutura do protocolo IPv6 através da captura de pacotes com o programa Wireshark, onde poderemos observar melhor os tópicos aprendidos durante a aula teórica.

Exercício - 1: Instalação de aplicativos

Para realizarmos os exercícios propostos neste laboratório, será necessário a instalação de alguns aplicativos em nossos nossos noteboks, como:

- Wireshark
- cliente SSH (putty, por exemplo).

Você pode consultar seu buscador preferido.

Exercício 0: Configuração nativa de IPv6 em seu notebook.

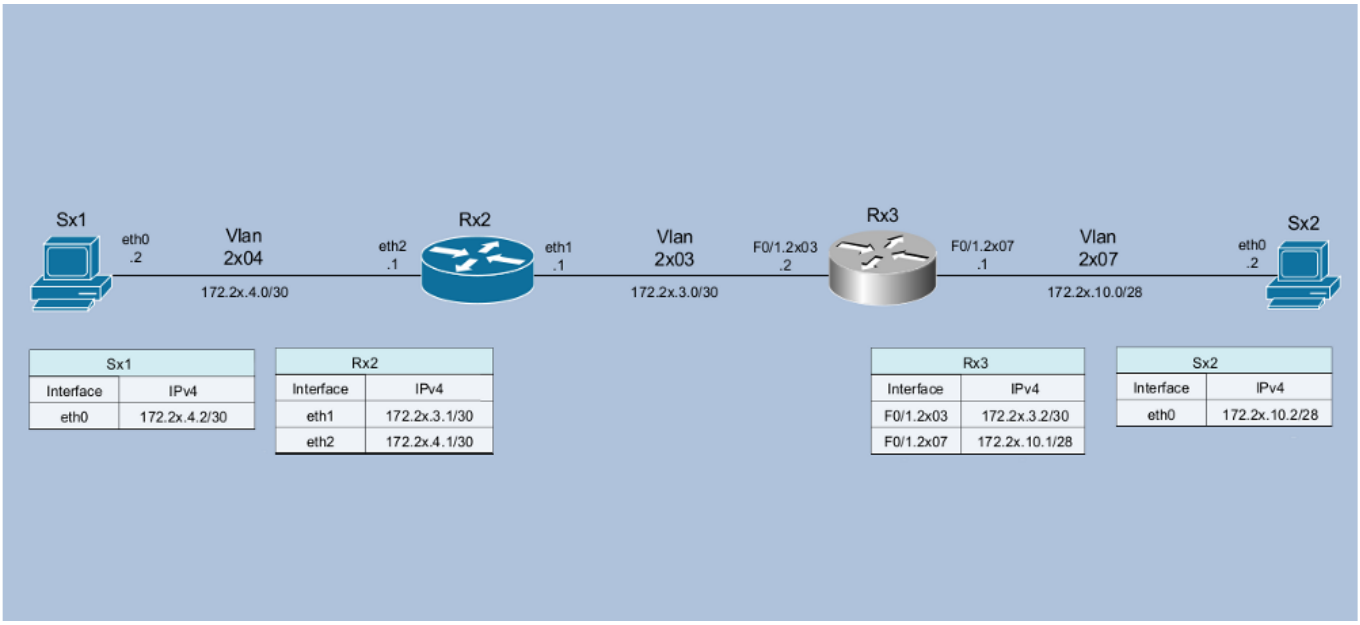
Consulte no site http://ipv6.br o artigo “Habilitando IPv6 em Sistemas Operacionais”. Nele você encontrará como configurar o IPv6 nos principais Sistemas Operacionais.

Configure...

Acesse sites ipv6, faça pings e traceroutes em IPv6.

Dê um traceroute para www.google.com.br e para www6.terra.com.br

Exercício 1: Acesso e configurações de rede.



A partir de agora, todos os exercícios do laboratório seguirão o seguinte cenário:

- A turma será dividida em grupos, onde cada grupo representara um AS. Inicialmente esse AS será composto por dois servidores e dois roteadores, um Cisco e um Linux/Quagga.
- Para acessar o roteador CISCO do laboratório, é necessário fazer um ssh para o endereço xxx.xxx.xxx.xxx, ou xxxx:xxxx:x:xxxx::xxx, na porta 3443, com usuário “labnicX” e senha “labgrupoX” (sem as aspas), onde X representa o número do grupo.

```
moreiras@atenas:~$ ssh xxxx:xxxx:x:xxxx::xxx -p3443 -llabnicX
The authenticity of host '[xxxx:xxxx:x:xxxx::xxx]:3443
([xxxx:xxxx:x:xxxx::xxx]:3443)' can't be established.
RSA key fingerprint is 7d:af:21:68:6f:9b:13:cd:9d:ce:07:b5:b0:4e:40:e5.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[xxxx:xxxx:x:xxxx::xxx]:3443' (RSA) to the list of known
hosts.
labnicX@xxxx:xxxx:x:xxxx::xxx's password: labg
Last login: Mon Jun 15 02:23:42 2009 from atenas.ceptro.br
You are in a limited shell.
Type '?' or 'help' to get the list of allowed commands
labnicX:~$
```

Com isso você estará logado no servidor de administração de nosso laboratório. À partir dele, você poderá acessar os servidores e roteadores disponíveis. O usuário labnicX dá acesso à uma sessão limitada, onde é possível executar apenas os comandos para acessar os componentes do laboratório:

```
labnicX:~$ help
console  exit  help  juniper  router  rx1  rx2  rx3  rx4  server  sx1  sx2  sx3
labnicX:~$
```

Os comandos “server x1” e “server x2” dão acesso aos servidores, onde x é o número do grupo. O comando “router x3” dá acesso ao roteador Cisco: use o usuário “cisco” e senha “cisco”. Por fim, o comando “router x2” dá acesso ao roteador Linux/Quagga. Use a senha “labgrupoX” para os servidores e roteadores Linux/Quagga.

O grupo deve testar o acesso a todos eles.

```
labnicX:~$
labnicX:~$ server x1 (troque o x pelo número do seu grupo)
Senha: labgrupoX
entered into CT 110
[root@SX1 /]# exit
logout
exited from CT 110

labnicX:~$
labnicX:~$ router x2 (troque o x pelo número do seu grupo)
entered into CT 112
[root@RX2 /]# exit
logout
exited from CT 112

labnicX:~$
labnicX:~$ router x3 (troque o x pelo número do seu grupo)
Trying 192.168.50.1...
Connected to 192.168.50.1.
Escape character is '^]'.
User Access Verification
Username: cisco
Password: cisco
router-RX3#
router-RX3#exit
Connection closed by foreign host.

labnicX:~$
labnicX:~$ server x2 (troque o x pelo número do seu grupo)
entered into CT 114
[root@SX2 /]# exit
logout
exited from CT 114
labnicX:~$
```

Nessa fase do laboratório, apenas o IPv4 está configurado, embora todos os equipamentos sejam capazes de executar IPv6. Não são usados protocolos de roteamento inicialmente, apenas rotas estáticas. Verifique as configurações de rede e execute testes de conectividade entre todos os elementos do laboratório. Use, por exemplo, comandos como “ip”, “ping”, “traceroute”, “mtr”, etc. Procure entender como estão configurados os equipamentos e as rotas.

Se não houver conectividade entre todos os elementos, avise aos instrutores e tente descobrir onde está o problema e resolvê-lo.

Exercício 2: Captura e análise de pacotes.

Para a captura de pacotes nos servidores e roteadores Linux, será utilizado o comando “tcpdump”. Para as análises será utilizado o programa “Wireshark”, previamente instalado nos notebook dos participantes do curso.

Para nos habituarmos ao uso das ferramentas, vamos monitorar o tráfego na interface eth2, do roteador Rx2, e executar um traceroute, de Sx1 para Sx2.

- No router Rx2:

```
labnicX:~$ router x2
Senha:
entered into CT 112
[root@RX2 /]#
[root@RX2 /]# tcpdump -i eth2 -s 0 -w /captura/exerc02.pcap
tcpdump: listening on eth2, link-type EN10MB (Ethernet), capture size 65535 bytes
```

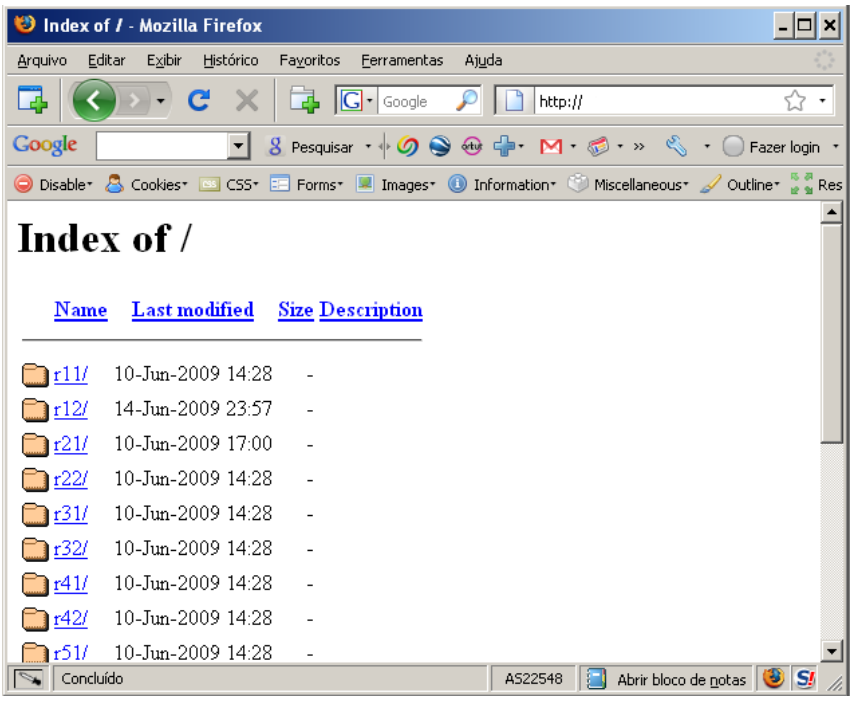
- No servidor Sx1:

```
labnicX:~$ server x1
Senha:
entered into CT 110
[root@SX1 /]# traceroute 172.2X.10.2
traceroute to 172.2X.10.2 (172.2X.10.2), 30 hops max, 46 byte packets
 1  172.2X.4.1 (172.2X.4.1)  3.027 ms  0.026 ms  0.025 ms
 2  172.2X.3.2 (172.2X.3.2)  0.642 ms  0.663 ms  0.651 ms
 3  172.2X.10.2 (172.2X.10.2)  1.851 ms  0.277 ms  0.275 ms
[root@SX1 /]#
```

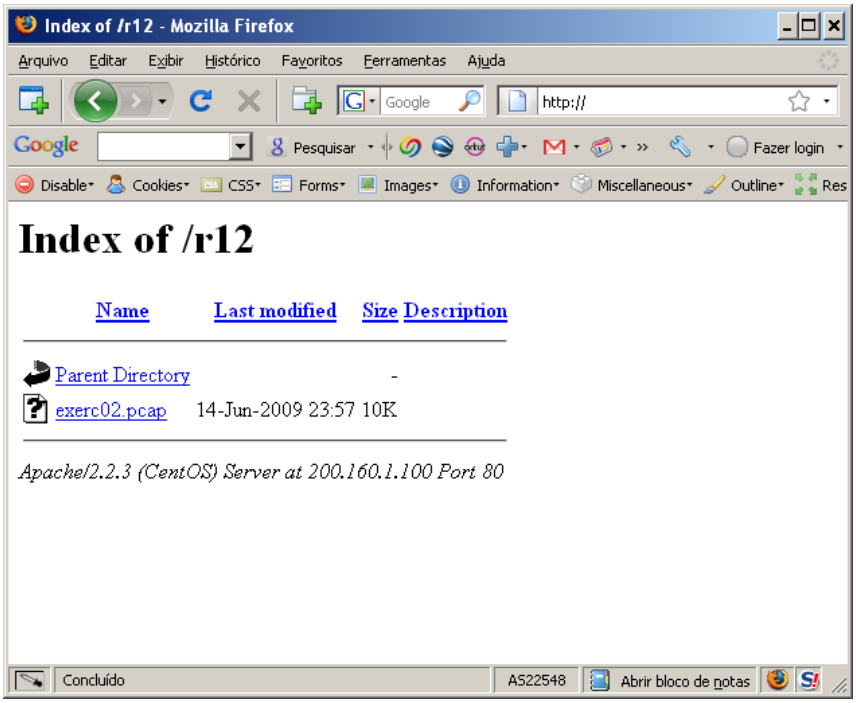
- Novamente, no router Rx2:

```
[root@RX2 /]# tcpdump -i eth2 -s 0 -w /captura/exerc02.pcap
tcpdump: listening on eth2, link-type EN10MB (Ethernet), capture size 65535 bytes
[CTRL + C]
127 packets captured
127 packets received by filter
0 packets dropped by kernel
[root@RX2 /]#
```

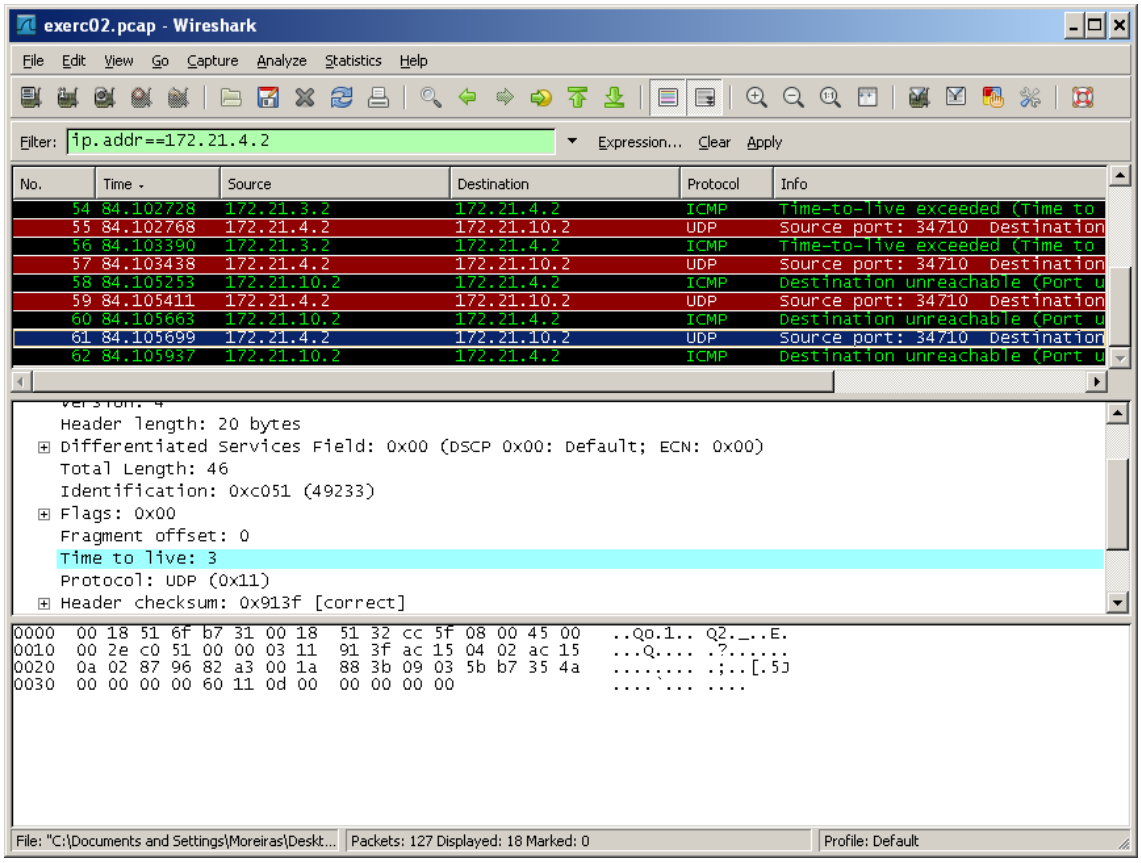
Em cerca de 1 minuto, no máximo, um script copiará este arquivo para um diretório compartilhado via Web em nosso servidor de gerenciamento. Aguarde alguns instantes e, usando um navegador em seu notebook, acesse o endereço http://[xxxx:xxxx:x:xxxx::xxx]/captura/ (o mesmo utilizado com o ssh).



Entre na pasta correspondente ao router x2 e salve o arquivo exerc02.pcap em seu notebook.



Abra o arquivo no Wireshark.
Aplique o filtro ip.addr==''endereço de origem do traceroute", se quiser, para facilitar a visualização, e responda as seguintes 2 questões:
1 – Qual protocolo é utilizado para o envio das mensagens pela origem?
2 – Quantos pacotes são enviados para cada valor de TTL?


exerc02.pcap - Wireshark
File Edit View Go Capture Analyze Statistics Help
Filter: ip.addr==172.21.4.2
Expression... Clear Apply
No. Time Source Destination Protocol Info
54 84.102728 172.21.3.2 172.21.4.2 ICMP Time-to-live exceeded (Time to
55 84.102768 172.21.4.2 172.21.10.2 UDP Source port: 34710 Destination
56 84.103390 172.21.3.2 172.21.4.2 ICMP Time-to-live exceeded (Time to
57 84.103438 172.21.4.2 172.21.10.2 UDP Source port: 34710 Destination
58 84.105253 172.21.10.2 172.21.4.2 ICMP Destination unreachable (Port u
59 84.105411 172.21.4.2 172.21.10.2 UDP Source port: 34710 Destination
60 84.105663 172.21.10.2 172.21.4.2 ICMP Destination unreachable (Port u
61 84.105699 172.21.4.2 172.21.10.2 UDP Source port: 34710 Destination
62 84.105937 172.21.10.2 172.21.4.2 ICMP Destination unreachable (Port u
Header length: 20 bytes
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
Total Length: 46
Identification: 0xc051 (49233)
Flags: 0x00
Fragment offset: 0
Time to live: 3
Protocol: UDP (0x11)
Header checksum: 0x913f [correct]
0000 00 18 51 6f b7 31 00 18 51 32 cc 5f 08 00 45 00 ..Qo.1.. Q2._..E.
0010 00 2e c0 51 00 00 03 11 91 3f ac 15 04 02 ac 15 ...Q.... .?.....
0020 0a 02 87 96 82 a3 00 1a 88 3b 09 03 5b b7 35 4a .....;..[.5J
0030 00 00 00 00 60 11 0d 00 00 00 00 00 ....
File: "C:\Documents and Settings\Moreiras\Deskt... Packets: 127 Displayed: 18 Marked: 0 Profile: Default

Exercício 3: IPv6 – endereços locais.

Habilite o IPv6 nos equipamentos e verifique que, mesmo que os endereços IPv6 ainda não tenha sido configurados, já há endereços do tipo “link-local” em cada um deles. Pode ser que para alguns dos equipamentos, o IPv6 já esteja habilitado.

Exemplo, no Linux:

```
[root@SX1 /]# ip addr show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue state UNKNOWN
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
    inet6 ::1/128 scope host
       valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen
1000
    link/ether 00:18:51:32:CC:5F brd ff:ff:ff:ff:ff:ff
    inet 172.2x.4.2/30 brd 172.2x.4.3 scope global eth0
    inet6 fe80::218:51ff:fe32:cc5f/64 scope link
       valid_lft forever preferred_lft forever
```

Nesse exemplo, o IPv6 já está habilitado (nas máquinas do laboratório, o IPv6 foi incluído no kernel; uma alternativa seria usá-lo como um módulo, carregado ou não por padrão). No caso do ipv6 ter sido compilado como módulo e não ser carregado por padrão, é necessário usar o comando “modprobe ipv6”.

Exemplo, no Cisco:

```
router-RX3#show ipv6 interface FastEthernet 0/1.2x03
router-RX3#
```

Nesse caso, o IPv6 não está habilitado. É necessário habilitá-lo:

```
router-RX3#
router-RX3#configure terminal
router-RX3(config)#interface FastEthernet 0/1.2x03
router-RX3(config-subif)#ipv6 enable
router-RX3(config-subif)#end
router-RX3#show ipv6 interface FastEthernet 0/1.2x03
FastEthernet0/1.2x03 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::224:97FF:FEC1:C8BD
  No Virtual link-local address(es):
  Description: Conexao-RX2
  No global unicast address is configured
  Joined group address(es):
    FF02::1
    FF02::1:FFC1:C8BD
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ICMP unreachables are sent
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds (using 22434)
router-RX3#
```

Verifique que com os endereços “link-local” já há conectividade IPv6 para cada segmento de rede. E que não há conectividade entre diferentes seguimentos.

Exemplo:

```
[root@SX1 /]# ping6 fe80::218:51ff:fe32:cc5f
64 bytes from fe80::218:51ff:fe32:cc5f: icmp_seq=0 ttl=64 time=2.25 ms
64 bytes from fe80::218:51ff:fe32:cc5f: icmp_seq=1 ttl=64 time=0.079 ms
64 bytes from fe80::218:51ff:fe32:cc5f: icmp_seq=2 ttl=64 time=0.088 ms
```

Suas saídas se parecem mais com esta (abaixo)? O que está faltando no comando?

```
[root@SX1 /]# ping6  fe80::218:51ff:fe32:cc5f
connect: Invalid argument
```

Descubra o endereço físico (MAC) de cada interface e responda a seguinte questão: Como os endereços “link-local” são formados à partir dos endereços físicos?

Exercício 4: IPv6 – analisando o cabeçalho dos pacotes.

Neste exercício, vamos analisar o cabeçalho do protocolo IPv6, e tentar descobrir algumas diferenças em relação ao IPv4.

Vamos capturar os pacotes de pings, v4 e v6, enviados de Sx1 para Rx2, no roteador Rx2.

- No router Rx2:

```
[root@RX2 ~]# ip addr show eth2
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:18:51:6F:B7:31 brd ff:ff:ff:ff:ff:ff
    inet 172.2x.4.1/30 brd 172.2x.4.3 scope global eth2
    inet6 fe80::218:51ff:fe6f:b731/64 scope link
       valid_lft forever preferred_lft forever

[root@RX2 ~]# tcpdump -i eth2 -s 0 -w /captura/exerc04.pcap
tcpdump: listening on eth2, link-type EN10MB (Ethernet), capture size 65535 bytes
```

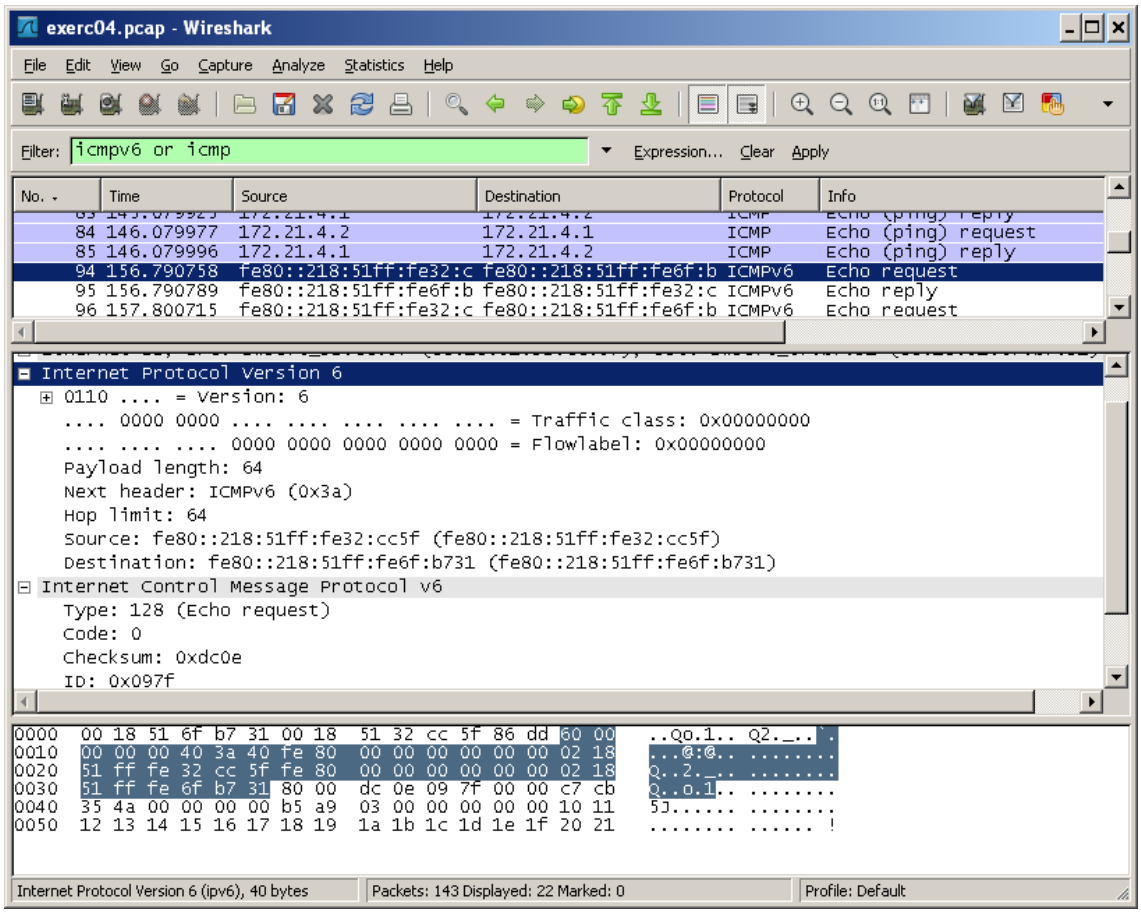
- No servidor Sx1:

```
[root@SX1 /]# ping -c 5 172.2x.4.1
PING 172.2x.4.1 (172.2x.4.1) 56(84) bytes of data.
64 bytes from 172.2x.4.1: icmp_seq=0 ttl=64 time=2.06 ms
64 bytes from 172.2x.4.1: icmp_seq=1 ttl=64 time=0.031 ms
64 bytes from 172.2x.4.1: icmp_seq=2 ttl=64 time=0.081 ms
64 bytes from 172.2x.4.1: icmp_seq=3 ttl=64 time=0.038 ms
64 bytes from 172.2x.4.1: icmp_seq=4 ttl=64 time=0.044 ms

--- 172.2x.4.1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4002ms
rtt min/avg/max/mdev = 0.031/0.451/2.061/0.805 ms, pipe 2
[root@SX1 /]# ping6 -c 5 fe80::218:51ff:fe6f:b731 -I eth0
PING fe80::218:51ff:fe6f:b731(fe80::218:51ff:fe6f:b731) from fe80::218:51ff:fe32:cc5f eth0:
56 data bytes
64 bytes from fe80::218:51ff:fe6f:b731: icmp_seq=0 ttl=64 time=0.061 ms
64 bytes from fe80::218:51ff:fe6f:b731: icmp_seq=1 ttl=64 time=0.085 ms
64 bytes from fe80::218:51ff:fe6f:b731: icmp_seq=2 ttl=64 time=0.035 ms
64 bytes from fe80::218:51ff:fe6f:b731: icmp_seq=3 ttl=64 time=0.089 ms
64 bytes from fe80::218:51ff:fe6f:b731: icmp_seq=4 ttl=64 time=0.036 ms

--- fe80::218:51ff:fe6f:b731 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4010ms
rtt min/avg/max/mdev = 0.035/0.061/0.089/0.023 ms, pipe 2
```

Abra o arquivo no Wireshark, em seu notebook.
Você pode utilizar o seguinte filtro, para facilitar a visualização: “icmp or icmpv6”



Compare os pacotes IPv4 e IPv6... Identifique cada um dos campos do cabeçalho IP nos dois casos, e observe seus valores. Compare também o ICMP. Responda às seguintes questões?

- Qual a diferença de tamanho entre o cabeçalho IPv4 e o cabeçalho IPv6?
- Há diferenças também no cabeçalho ICMP? Quais?

Exercício 5: IPv6 – cabeçalhos de extensão.
Neste exercício, vamos verificar a existência dos cabeçalhos de extensão, gerando a necessidade de fragmentação no comando ping.

Vamos capturar os pacotes de pings, v4 e v6, enviados de Sx1 para Rx2, no roteador Rx2, como no exercício anterior. Mas vamos especificar 2000 bytes para o tamanho do pacote.

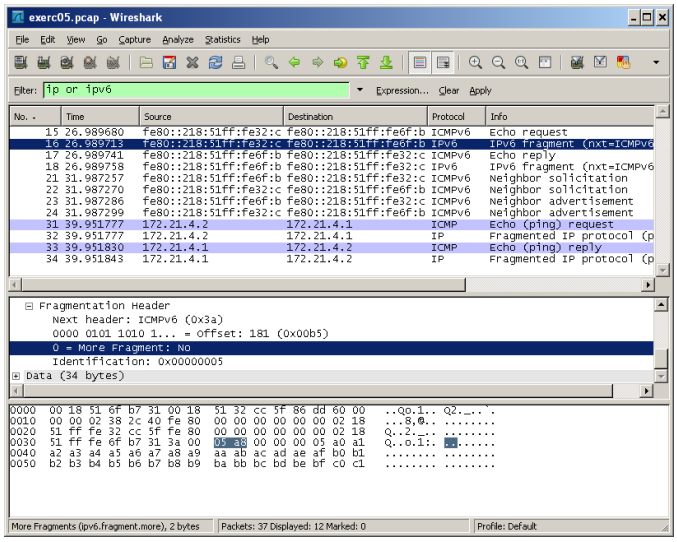
- No router Rx2:

```
[root@RX2 ~]# tcpdump -i eth2 -s 0 -w /captura/exerc05.pcap
tcpdump: listening on eth2, link-type EN10MB (Ethernet), capture size 65535 bytes
37 packets captured
37 packets received by filter
0 packets dropped by kernel
```

- No Servidor Sx1:

```
[root@SX1 /]# ping6 -c 1 -s 2000 fe80::218:51ff:fe6f:b731 -I eth0
PING fe80::218:51ff:fe6f:b731(fe80::218:51ff:fe6f:b731) from
fe80::218:51ff:fe32:cc5f eth0: 2000 data bytes
2008 bytes from fe80::218:51ff:fe6f:b731: icmp_seq=0 ttl=64 time=0.112 ms
--- fe80::218:51ff:fe6f:b731 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.112/0.112/0.112/0.000 ms, pipe 2
[root@SX1 /]# ping -c 1 -s 2000 172.2x.4.1
PING 172.2x.4.1 (172.2x.4.1) 2000(2028) bytes of data.
2008 bytes from 172.2x.4.1: icmp_seq=0 ttl=64 time=1.38 ms
--- 172.2x.4.1 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 1.388/1.388/1.388/0.000 ms, pipe 2
[root@SX1 /]#
```

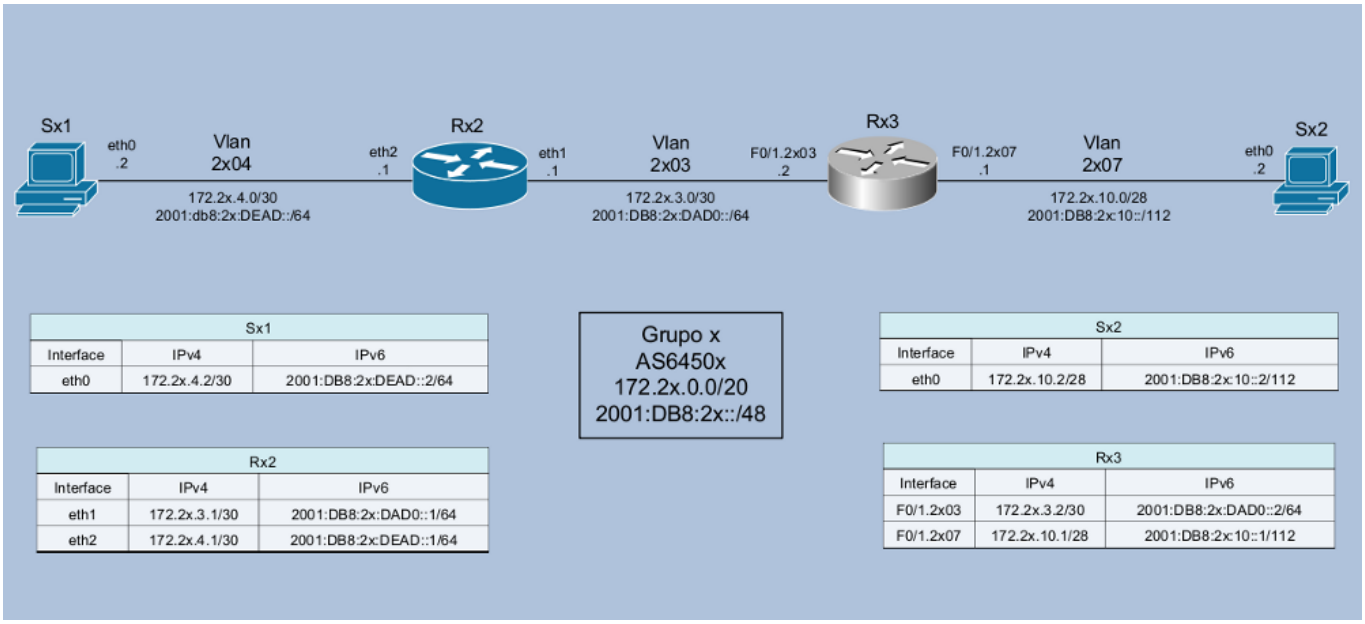
Abra o arquivo no Wireshark, e utilize o filtro “ipv6 or ip” para facilitar a visualização.



- Verifique a existência do cabeçalho de fragmentação e responda às seguintes questões?
- Qual a diferença entre o processo de fragmentação no IPv4 e no IPv6?
- Qual o tamanho do cabeçalho de extensão (fragmentação)?
- Qual a diferença do valor do campo Next Header no cabeçalho v6 do exercício 04 para este exercício?

Exercício 6a: IPv6 – configuração manual dos endereços.

Neste exercício, vamos configurar os endereços de nosso bloco (2001:db8:2x::/48), conforme a figura à seguir.



- No servidor Sx1:

```
[root@SX1 ]# ip -6 addr add 2001:db8:2x:dead::2/64 dev eth0
[root@SX1 ]# ip addr show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen
1000
    link/ether 00:18:51:32:CC:5F brd ff:ff:ff:ff:ff:ff
    inet 172.2x.4.2/30 brd 172.2x.4.3 scope global eth0
    inet6 2001:db8:2x:dead::2/64 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80::218:51ff:fe32:cc5f/64 scope link
       valid_lft forever preferred_lft forever
```

- No servidor Sx2:

```
[root@SX2 /]# ip -6 addr add 2001:db8:2x:10::2/112 dev eth0
[root@SX2 /]# ip addr show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen
1000
    link/ether 00:18:51:F2:87:5B brd ff:ff:ff:ff:ff:ff
    inet 172.2x.10.2/28 brd 172.2x.10.15 scope global eth0
    inet6 2001:db8:2x:10::2/112 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80::218:51ff:fef2:875b/64 scope link
       valid_lft forever preferred_lft forever
```

- No roteador Rx2:

```
[root@RX2 ~]# ip -6 addr add 2001:db8:2x:dad0::1/64 dev eth1
[root@RX2 ~]# ip -6 addr add 2001:db8:2x:dead::1/64 dev eth2
[root@RX2 ~]# ip addr show eth1
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen
1000
    link/ether 00:18:51:1D:41:8A brd ff:ff:ff:ff:ff:ff
    inet 172.2x.3.1/30 brd 172.2x.3.3 scope global eth1
    inet6 2001:db8:2x:dad0::1/64 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80::218:51ff:fe1d:418a/64 scope link
       valid_lft forever preferred_lft forever
[root@RX2 ~]# ip addr show eth2
4: eth2: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen
1000
    link/ether 00:18:51:6F:B7:31 brd ff:ff:ff:ff:ff:ff
    inet 172.2x.4.1/30 brd 172.2x.4.3 scope global eth2
    inet6 2001:db8:2x:dead::1/64 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80::218:51ff:fe6f:b731/64 scope link
       valid_lft forever preferred_lft forever
```

- No roteador Rx3:

```
router-RX3#
router-RX3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
router-RX3(config)#interface fastEthernet 0/1.2x03
router-RX3(config-subif)#ipv6 address 2001:db8:2x:dad0::2/64
router-RX3(config-subif)#end
router-RX3#show ipv6 interface FastEthernet 0/1.2x03
FastEthernet0/1.2x03 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::224:97FF:FEC1:C8BD
  No Virtual link-local address(es):
  Description: Conexao-RX2
  Global unicast address(es):
    2001:DB8:2x:DAD0::2, subnet is 2001:DB8:2x:DAD0::/64
  Joined group address(es):
    FF02::1
    FF02::1:FF00:2
    FF02::1:FFC1:C8BD
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ICMP unreachables are sent
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds (using 22434)
router-RX3#

router-RX3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
router-RX3(config)#interface fastEthernet 0/1.2x07
router-RX3(config-subif)#ipv6 address 2001:db8:2x:10::1/112
router-RX3(config-subif)#end
router-RX3#show ipv6 interface FastEthernet 0/1.2x07
```

```
FastEthernet0/1.2x07 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::224:97FF:FEC1:C8BD
  No Virtual link-local address(es):
  Description: Conexao-SX2
  Global unicast address(es):
    2001:DB8:2x:10::1, subnet is 2001:DB8:2x:10::/112
  Joined group address(es):
    FF02::1
    FF02::1:FF00:1
    FF02::1:FFC1:C8BD
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ICMP unreachables are sent
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds (using 26577)
router-R13#
```

Verifique que agora há conectividade local, através dos novos endereços, válidos globalmente, bem como dos pré-existentes endereços “link-local”. Não há conectividade entre as diferentes redes, no entanto, porque as rotas ainda não foram configuradas.

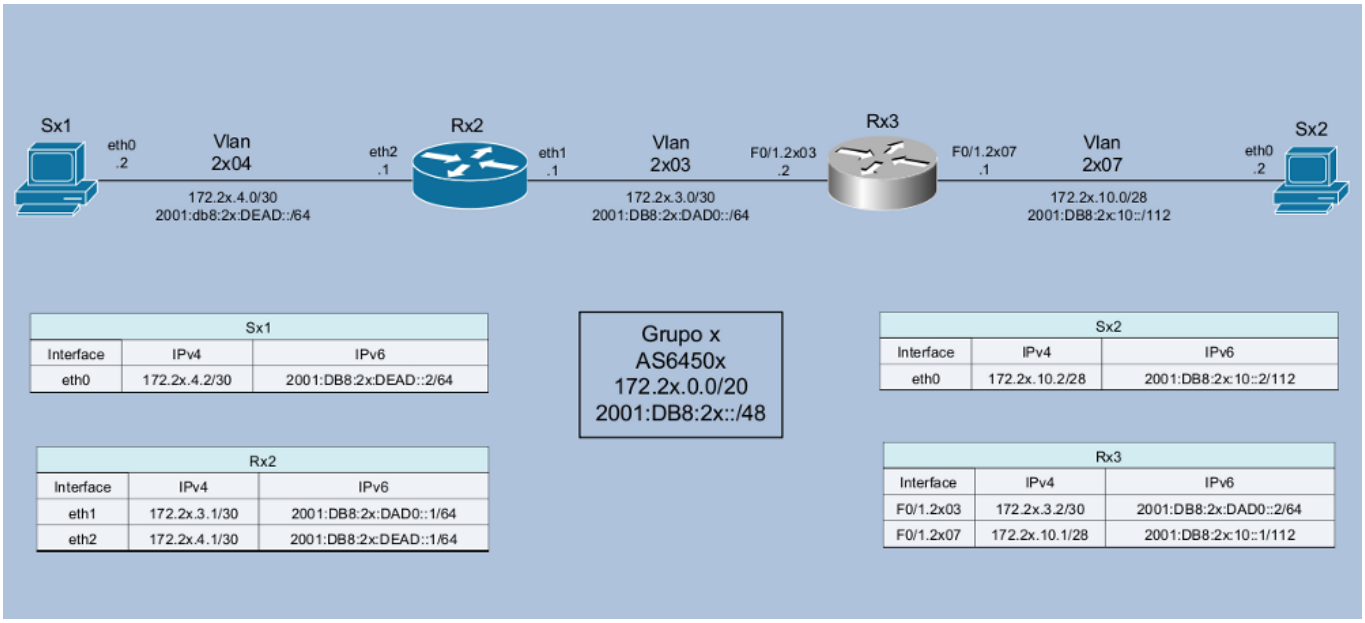
Veja que é possível adicionar outros endereços IPv6 às interfaces.

Experimente, por exemplo, adicionar novos endereços à Sx1:

```
[root@SX1 ~]# ip -6 addr add 2001:db8:2x:dead::60:61e/64 dev eth0
[root@SX1 ~]# ip -6 addr add 2001:db8:2x:dead::cafe:dad0/64 dev eth0
[root@SX1 ~]# ip addr show eth0
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 00:18:51:32:CC:5F brd ff:ff:ff:ff:ff:ff
    inet 172.2x.4.2/30 brd 172.2x.4.3 scope global eth0
    inet6 2001:db8:2x:dead::2/64 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80::218:51ff:fe32:cc5f/64 scope link
       valid_lft forever preferred_lft forever
    inet6 2001:db8:2x:dead::cafe:dad0/64 scope global
       valid_lft forever preferred_lft forever
    inet6 2001:db8:2x:dead::60:61e/64 scope global
       valid_lft forever preferred_lft forever

[root@SX1 ~]# ip -6 addr del 2001:db8:2x:dead::60:61e/64 dev eth0
[root@SX1 ~]# ip -6 addr del 2001:db8:2x:dead::cafe:dad0/64 dev eth0
```

Exercício 6b: IPv6 – configuração manual das rotas.
Neste exercício, vamos configurar as rotas, manualmente, de forma a ter conectividade v4 e v6 em nosso laboratório.



Procure observar a configuração IPv4 e "copiá-la" para o contexto IPv6, antes de consultar os exemplos que seguem.

Exs:

- Para Sx1:

```
[root@SX1 ~]# ip route add default via 2001:db8:2x:dead::1
[root@SX1 ~]# ip -6 route show
2001:db8:2x:dead::/64 dev eth0  metric 256  expires 21331916sec mtu 1500 advmss
1440 hoplimit 4294967295
fe80::/64 dev eth0  metric 256  expires 21296278sec mtu 1500 advmss 1440 hoplimit
4294967295
default via 2001:db8:2x:dead::1 dev eth0  metric 1024  expires 21334251sec mtu 1500
advmss 1440 hoplimit 4294967295
unreachable default dev lo  proto none  metric -1  error -101 hoplimit 255
ff00::/8 dev eth0  metric 256  expires 21296278sec mtu 1500 advmss 1440 hoplimit
4294967295
unreachable default dev lo  proto none  metric -1  error -101 hoplimit 255
[root@SX1 ~]#
```

- Para Sx2:

```
[root@SX2 /]# ip route add default via 2001:db8:2x:10::1
[root@SX2 /]# ip -6 route show
2001:db8:2x:10::/112 dev eth0  metric 256  expires 21332280sec mtu 1500 advmss 1440 hoplimit
4294967295
fe80::/64 dev eth0  metric 256  expires 21296188sec mtu 1500 advmss 1440 hoplimit 4294967295
default via 2001:db8:2x:10::1 dev eth0  metric 1024  expires 21334371sec mtu 1500 advmss 1440
hoplimit 4294967295
unreachable default dev lo  proto none  metric -1  error -101 hoplimit 255
ff00::/8 dev eth0  metric 256  expires 21296188sec mtu 1500 advmss 1440 hoplimit 4294967295
unreachable default dev lo  proto none  metric -1  error -101 hoplimit 255
[root@SX2 /]#
```

- Para Rx2:

```
[root@RX2 ~]# ip route add default via 2001:db8:2x:dad0::2
[root@RX2 ~]# ip -6 route show
2001:db8:2x:dad0::/64 dev eth1  metric 256  expires 21334232sec mtu 1500 advmss
1440 hoplimit 4294967295
2001:db8:2x:dead::/64 dev eth2  metric 256  expires 21334251sec mtu 1500 advmss
1440 hoplimit 4294967295
2001:db8:2x:faca::/64 dev eth0  metric 256  expires 21334205sec mtu 1500 advmss
1440 hoplimit 4294967295
fe80::/64 dev eth0  metric 256  expires 21295857sec mtu 1500 advmss 1440 hoplimit
4294967295
fe80::/64 dev eth1  metric 256  expires 21295863sec mtu 1500 advmss 1440 hoplimit
4294967295
fe80::/64 dev eth2  metric 256  expires 21295868sec mtu 1500 advmss 1440 hoplimit
4294967295
unreachable fe80::/64 dev lo  metric 256  expires 21295873sec error -101 mtu 16436
advmss 16376 hoplimit 4294967295
default via 2001:db8:2x:dad0::2 dev eth1  metric 1024  expires 21334364sec mtu 1500
advmss 1440 hoplimit 4294967295
unreachable default dev lo  proto none  metric -1  error -101 hoplimit 255
ff00::/8 dev eth0  metric 256  expires 21295857sec mtu 1500 advmss 1440 hoplimit
4294967295
ff00::/8 dev eth1  metric 256  expires 21295863sec mtu 1500 advmss 1440 hoplimit
4294967295
ff00::/8 dev eth2  metric 256  expires 21295868sec mtu 1500 advmss 1440 hoplimit
4294967295
unreachable ff00::/8 dev lo  metric 256  expires 21295873sec error -101 mtu 16436
advmss 16376 hoplimit 4294967295
unreachable default dev lo  proto none  metric -1  error -101 hoplimit 255
```

- Para Rx3:

```
router-RX3#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
router-RX3(config)#ipv6 unicast-routing
router-RX3(config)#ipv6 route ::0/0 2001:db8:2x:dad0::1
router-RX3(config)#end
router-RX3#show ipv6 route
IPv6 Routing Table - Default - 6 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, M - MIPv6, R - RIP, I1 - ISIS L1
       I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
       EX - EIGRP external
       O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
S   ::/0 [1/0]
     via 2001:DB8:2x:DAD0::1
C   2001:DB8:2x:10::/112 [0/0]
     via FastEthernet0/1.2x07, directly connected
L   2001:DB8:2x:10::1/128 [0/0]
     via FastEthernet0/1.2x07, receive
C   2001:DB8:2x:DAD0::/64 [0/0]
     via FastEthernet0/1.2x03, directly connected
L   2001:DB8:2x:DAD0::2/128 [0/0]
     via FastEthernet0/1.2x03, receive
L   FF00::/8 [0/0]
     via Null0, receive
```

Após as configurações, teste a conectividade ponta a ponta, com ping6 de Sx1 para Sx2.

Ex:

```
[root@SX1~]# ping6 -c 5 2001:db8:2x:10::2
PING 2001:db8:2x:10::2(2001:db8:2x:10::2) 56 data bytes
64 bytes from 2001:db8:2x:10::2: icmp_seq=0 ttl=62 time=0.441 ms
64 bytes from 2001:db8:2x:10::2: icmp_seq=1 ttl=62 time=0.458 ms
64 bytes from 2001:db8:2x:10::2: icmp_seq=2 ttl=62 time=0.454 ms
64 bytes from 2001:db8:2x:10::2: icmp_seq=3 ttl=62 time=0.408 ms
64 bytes from 2001:db8:2x:10::2: icmp_seq=4 ttl=62 time=0.456 ms

--- 2001:db8:2x:10::2 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4009ms
rtt min/avg/max/mdev = 0.408/0.443/0.458/0.026 ms, pipe 2
[root@SX1 ~]# traceroute6 2001:db8:2x:10::2
traceroute to 2001:db8:2x:10::2 (2001:db8:2x:10::2) from 2001:db8:2x:dead::2, 30
hops max, 24 byte packets
 1  2001:db8:2x:dead::1 (2001:db8:2x:dead::1)  0.064 ms  0.035 ms  0.031 ms
 2  2001:db8:2x:dad0::2 (2001:db8:2x:dad0::2)  0.748 ms  0.603 ms  0.604 ms
 3  2001:db8:2x:10::2 (2001:db8:2x:10::2)  0.371 ms  0.313 ms  0.339 ms
[root@SX1 ~]#
```

Exercício 7: IPv6 – Neighbour Discovery.

Neste exercício, vamos observar o funcionamento do protocolo Neighbour Discovery.

Vamos, primeiramente, limpar a tabela do Neighbour Discovery para a interface eth2 do roteador Rx2:

Verificação da tabela de vizinhos do IPv6

- No Linux:

```
[root@RX2]# ip -6 neighbor (esta tabela e similar a tabela ARP do IPv4)
```

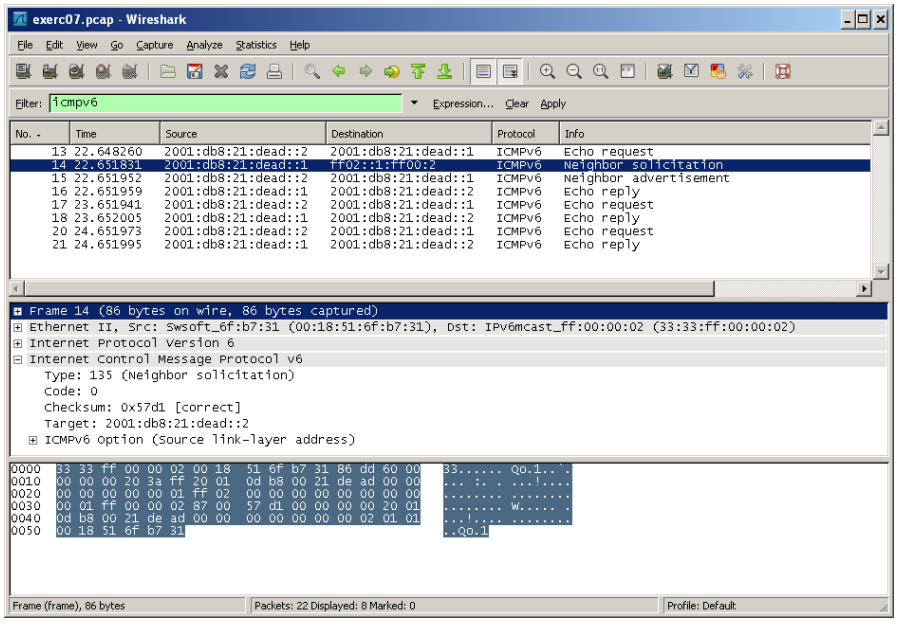
```
[root@RX2 ~]# ip neighbor flush dev eth2.
```

Vamos começar agora a capturar os pacotes, e pingar a interface à partir de Sx1... Logo após o ping paramos a captura.

```
[root@RX2 ~]# tcpdump -i eth2 -s 0 -w /captura/exerc07.pcap
tcpdump: listening on eth2, link-type EN10MB (Ethernet), capture size 65535 bytes
22 packets captured
22 packets received by filter
0 packets dropped by kernel
```

```
[root@SX1 ~]# ping6 -c 3 2001:db8:2x:dead::1
PING 2001:db8:2x:dead::1(2001:db8:2x:dead::1) 56 data bytes
64 bytes from 2001:db8:2x:dead::1: icmp_seq=0 ttl=64 time=3.72 ms
64 bytes from 2001:db8:2x:dead::1: icmp_seq=1 ttl=64 time=0.114 ms
64 bytes from 2001:db8:2x:dead::1: icmp_seq=2 ttl=64 time=0.040 ms
--- 2001:db8:21:dead::1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 0.040/1.291/3.721/1.718 ms, pipe 2
```

Abra o arquivo no Wireshark e use o filtro “icmpv6”. Verifique a existência das mensagens Neighbour Solicitation e Neighbour Advertisement.



Exercício 8: IPv6 – Path MTU Discovery.

Neste exercício, vamos observar o funcionamento do protocolo Path MTU Discovery. Primeiramente, vamos diminuir “artificialmente” o MTU de uma das interface eth1 de Rx2, e ativar o tcpdump na eth2:

```
[root@RX2 ~]# ip link set eth1 mtu 1280
[root@RX2 ~]# ip addr show eth1
3: eth1: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1280 qdisc pfifo_fast state UP qlen
1000
    link/ether 00:18:51:1D:41:8A brd ff:ff:ff:ff:ff:ff
    inet 172.2x.3.1/30 brd 172.2x.3.3 scope global eth1
    inet6 2001:db8:2x:dad0::1/64 scope global
       valid_lft forever preferred_lft forever
    inet6 fe80::218:51ff:fe1d:418a/64 scope link
       valid_lft forever preferred_lft forever
[root@RX2 ~]# tcpdump -i eth2 -s 0 -w /captura/exerc08.pcap
tcpdump: listening on eth1, link-type EN10MB (Ethernet), capture size 65535 bytes
```

Vamos, então, executar um ping, de Sx1 para Sx2, com tamanho de pacote maior que o MTU especificado, e observar o que ocorre.

```
[root@SX1 ~]# ping6 -c 5 -s 1500 2001:db8:2x:10::2
PING 2001:db8:2x:10::2(2001:db8:2x:10::2) 1500 data bytes
From 2001:db8:2x:dead::1 icmp_seq=0 Packet too big: mtu=1280
1508 bytes from 2001:db8:2x:10::2: icmp_seq=1 ttl=62 time=1.39 ms
1508 bytes from 2001:db8:2x:10::2: icmp_seq=2 ttl=62 time=1.29 ms
1508 bytes from 2001:db8:2x:10::2: icmp_seq=3 ttl=62 time=1.32 ms
1508 bytes from 2001:db8:2x:10::2: icmp_seq=4 ttl=62 time=1.33 ms
```

Paramos então a captura e analisamos os dados no Wireshark. Procure identificar a mensagem icmp de “Packet too big” e observar que tipo de informação extra ela traz. Responda à seguinte pergunta: qual o protocolo dessa mensagem, e o que você acha que ocorre se ela for bloqueada por um firewall?