

Tabelas Estáticas de Roteamento

Nome: Luiza Kuze Gomes

Disciplina: RCO786202

PARTE 1 - Tabelas estáticas de roteamento

IMUNES: pc0 (console) bash

```
root@pc0:/# ping 10.0.0.1
PING 10.0.0.1 (10.0.0.1) 56(84) bytes of data.
64 bytes from 10.0.0.1: icmp_seq=1 ttl=64 time=0.154 ms
64 bytes from 10.0.0.1: icmp_seq=2 ttl=64 time=0.066 ms
64 bytes from 10.0.0.1: icmp_seq=3 ttl=64 time=0.069 ms
^C
--- 10.0.0.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2082ms
rtt min/avg/max/mdev = 0.066/0.096/0.154/0.040 ms
root@pc0:/# ping 10.0.10.1
ping: connect: Network is unreachable
root@pc0:/# ping 10.0.10.2
ping: connect: Network is unreachable
root@pc0:/# █
```

5. Testes de conectividade de enlace e configuração do *default gateway*.

1. Por exemplo, no pc0 execute o comando: **ping 10.0.0.1**. Obteve sucesso? Sim ou não e por quê?

Sim, pois o pc0 também pertence a essa mesma sub-rede 10.0.0.1

2. Teste a conectividade do pc0 executando o comando: **ping 10.0.10.1**. Obteve sucesso? Sim ou não e por quê? Qual foi o erro observado?

Não, pois o pc0 não pertence a essa sub-rede. Erro foi “Network is unreachable”. Para funcionar é necessário configurar o roteador padrão.

3. Por exemplo, no pc0 execute o comando: **ping 10.0.10.2**. Obteve sucesso? Sim ou não e por quê? Qual foi o erro observado?

Não, pois o pc0 não pertence a essa sub-rede. Erro foi “Network is unreachable”. Para funcionar é necessário configurar o roteador padrão.

```
root@pc0:/# route add -net default gw 10.0.0.1
root@pc0:/# ping 10.0.10.1
PING 10.0.10.1 (10.0.10.1) 56(84) bytes of data.
64 bytes from 10.0.10.1: icmp_seq=1 ttl=64 time=0.102 ms
64 bytes from 10.0.10.1: icmp_seq=2 ttl=64 time=0.076 ms
64 bytes from 10.0.10.1: icmp_seq=3 ttl=64 time=0.065 ms
^C
--- 10.0.10.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2061ms
rtt min/avg/max/mdev = 0.065/0.081/0.102/0.015 ms
root@pc0:/# ping 10.0.10.2
PING 10.0.10.2 (10.0.10.2) 56(84) bytes of data.

^C
--- 10.0.10.2 ping statistics ---
6 packets transmitted, 0 received, 100% packet loss, time 5121ms
```

5. Teste novamente a conectividade, no pc0 execute o comando: ping 10.0.10.1 e ping 10.0.10.2. Obteve sucesso? O comportamento foi o mesmo das tentativas anteriores? Sim ou não e por quê? Qual foi o erro observado?

Dessa vez somente o ping de pc0 para 10.0.10.1 funcionou, pois com o comando “route add -net default gw 10.0.0.1” somente identificamos a interface do roteador (r0) diretamente conectado ao host (pc0).

6. Com os ping do item anterior ativos (um a cada tempo) rode o Wireshark no R0 (clique com o botão direito do mouse sobre o R0 e em seguida no menu wireshark eth0).

- i. Qual a origem e destino dos pacotes? Explique?
- ii. Qual a diferença no ping entre os dois itens?

ping 10.0.10.1

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.0.0.20	10.0.10.1	ICMP	98	Echo (ping) request id=0x...
2	0.000017	10.0.10.1	10.0.0.20	ICMP	98	Echo (ping) reply id=0x...
3	1.024001	10.0.0.20	10.0.10.1	ICMP	98	Echo (ping) request id=0x...

ping 10.0.10.2

1	0.000000	10.0.0.20	10.0.10.2	ICMP	98 Echo (ping) request	id=0x...
2	1.024180	10.0.0.20	10.0.10.2	ICMP	98 Echo (ping) request	id=0x...
3	2.048197	10.0.0.20	10.0.10.2	ICMP	98 Echo (ping) request	id=0x...
4	3.072174	10.0.0.20	10.0.10.2	ICMP	98 Echo (ping) request	id=0x...
5	4.096007	10.0.0.20	10.0.10.2	ICMP	98 Echo (ping) request	id=0x...

Para ambos os pings a origem é o 10.0.0.20 e o destino é o respectivo endereço do computador. A diferença é que apenas o segundo ping tem as requisições sendo respondidas.

6. Iniciando o roteamento.

1. Deixe o ping do do pc0 para o R1 e o wireshark - eth0 no R0 rodando e estabeleça uma rota no roteador R1 com o comando:

```
route add -net 10.0.0.0/24 gw 10.0.10.1
```

O que ocorreu com o ping e o wireshark? Por quê?

- Com este comando estamos: i) adicionando (*add*) uma rota ii) do tipo rede (*net*) iii) para a rede 10.0.0.0/24 iv) com o roteador (gw - *gateway*) v) 10.0.10.1 que identifica a interface do roteador, R0, diretamente conectado ao roteador R1.

33	16.256841	fe80::449c:3bff:fe4... ff02::2	ICMPv6	70 Router Solicitation from 46:9c:3b:46:09:8d	
34	16.383851	10.0.0.20	10.0.10.1	ICMP	98 Echo (ping) request id=0x000f, seq=35/8960, ttl=64 (reply in :)
35	16.383865	10.0.10.1	10.0.0.20	ICMP	98 Echo (ping) reply id=0x000f, seq=35/8960, ttl=64 (request in :)
36	16.768789	42:00:aa:00:00:00	42:00:aa:00:00:01	ARP	42 Who has 10.0.0.20? Tell 10.0.0.1
37	16.768842	42:00:aa:00:00:01	42:00:aa:00:00:00	ARP	42 10.0.0.20 is at 42:00:aa:00:00:01
38	17.407877	10.0.0.20	10.0.10.1	ICMP	98 Echo (ping) request id=0x000f, seq=36/9216, ttl=64 (reply in :)
39	17.407901	10.0.10.1	10.0.0.20	ICMP	98 Echo (ping) reply id=0x000f, seq=36/9216, ttl=64 (request in :)
40	18.431970	10.0.0.20	10.0.10.1	ICMP	98 Echo (ping) request id=0x000f, seq=37/9472, ttl=64 (reply in :)
41	18.431989	10.0.10.1	10.0.0.20	ICMP	98 Echo (ping) reply id=0x000f, seq=37/9472, ttl=64 (request in :)
42	19.455887	10.0.0.20	10.0.10.1	ICMP	98 Echo (ping) request id=0x000f, seq=38/9728, ttl=64 (reply in :)

O ping começou a funcionar após a configuração da rota, porque agora os pacotes podem ir e voltar entre o pc0 e o R1.

2. . Em todos os roteadores crie rotas para todas as redes. Em cada roteador deve-se criar 3 rotas, para as sub-redes "distantes", não diretamente conectadas. Lembre-se que os enlaces diretos já criam automaticamente rotas para as respectivas sub-redes diretamente conectadas ao equipamento, ou seja, entrega direta. Se tudo estiver correto, todos os PCs e roteadores devem pingar entre si.

- Crie rotas sempre pelo caminho mais curto, por exemplo, do R0 para a rede do pc1 e pc2 passando por R1 e para R2 respectivamente.

Executei o comando history para obter as rotas escolhidas.

```

root@R0:/# history
1 route add -net 10.0.1.0/24 gw 10.0.10.2
2 route add -net 10.0.2.0/24 gw 10.0.11.2
3 route add -net 10.0.12.0/30 gw 10.0.10.2

IMUNES: R1 (console) bash
root@R1:/# history
1 route add -net 10.0.0.0/24 gw 10.0.10.1
2 route add -net 10.0.2.0/24 gw 10.0.12.2
3 route add -net 10.0.11.0/30 gw 10.0.10.1

IMUNES: R2 (console) bash
root@R2:/# history
1 route add -net 10.0.0.0/24 gw 10.0.11.1
2 route add -net 10.0.1.0/24 gw 10.0.12.1
3 route add -net 10.0.10.0/30 gw 10.0.11.1

IMUNES: pc2 (console) bash
root@pc2:/# route add -net default gw 10.0.2.1
root@pc2:/#

IMUNES: pc1 (console) bash
root@pc1:/# route add -net default gw 10.0.1.1
root@pc1:/#

IMUNES: pc0 (console) bash
root@pc0:/# history
1 history
root@pc0:/# route add -net default gw 10.0.0.1

```

3. Trace e anote as rotas entre os *hosts* através do traceroute.

```

root@pc0:/# traceroute 10.0.1.20
traceroute to 10.0.1.20 (10.0.1.20), 30 hops max, 60 byte packets
1 10.0.0.1 (10.0.0.1) 0.095 ms 0.020 ms 0.026 ms
2 10.0.10.2 (10.0.10.2) 0.062 ms 0.046 ms 0.046 ms
3 10.0.1.20 (10.0.1.20) 0.052 ms 0.038 ms 0.038 ms
root@pc0:/# traceroute 10.0.2.20
traceroute to 10.0.2.20 (10.0.2.20), 30 hops max, 60 byte packets
1 10.0.0.1 (10.0.0.1) 0.070 ms 0.023 ms 0.013 ms
2 10.0.11.2 (10.0.11.2) 0.036 ms 0.023 ms 0.022 ms
3 10.0.2.20 (10.0.2.20) 0.044 ms 0.031 ms 0.030 ms
root@pc0:/#

IMUNES: pc1 (console) bash
root@pc1:/# traceroute 10.0.0.20
traceroute to 10.0.0.20 (10.0.0.20), 30 hops max, 60 byte packets
1 10.0.1.1 (10.0.1.1) 0.088 ms 0.022 ms 0.013 ms
2 10.0.10.1 (10.0.10.1) 0.033 ms 0.022 ms 0.021 ms
3 10.0.0.20 (10.0.0.20) 0.064 ms 0.038 ms 0.033 ms
root@pc1:/# traceroute 10.0.2.20
traceroute to 10.0.2.20 (10.0.2.20), 30 hops max, 60 byte packets
1 10.0.1.1 (10.0.1.1) 0.074 ms 0.020 ms 0.023 ms
2 10.0.12.2 (10.0.12.2) 0.056 ms 0.026 ms 0.024 ms
3 10.0.2.20 (10.0.2.20) 0.045 ms 0.033 ms 0.041 ms
root@pc1:/#

IMUNES: pc2 (console) bash
root@pc2:/# traceroute 10.0.0.20
traceroute to 10.0.0.20 (10.0.0.20), 30 hops max, 60 byte packets
1 10.0.2.1 (10.0.2.1) 0.068 ms 0.018 ms 0.013 ms
2 10.0.11.1 (10.0.11.1) 0.037 ms 0.020 ms 0.022 ms
3 10.0.0.20 (10.0.0.20) 0.045 ms 0.031 ms 0.028 ms
root@pc2:/# traceroute 10.0.1.20
traceroute to 10.0.1.20 (10.0.1.20), 30 hops max, 60 byte packets
1 10.0.2.1 (10.0.2.1) 0.074 ms 0.015 ms 0.012 ms
2 10.0.12.1 (10.0.12.1) 0.040 ms 0.020 ms 0.019 ms
3 10.0.1.20 (10.0.1.20) 0.036 ms 0.048 ms 0.027 ms
root@pc2:/#

```

7. Testando a queda de enlace. Com todas as rotas em perfeito funcionamento, gere um ping do pc0 para o pc2 e execute wireshark eth0 no R0 , em seguida "derrube" o enlace entre o R0 e R2. Por exemplo, no R2 execute o comando: `ifconfig eth1 down`. O que ocorreu com o ping e o wireshark? Por quê? Com este enlace comprometido qual seria a solução para a continuidade de funcionamento de toda a rede?

156	38.080027	10.0.0.20	10.0.2.20	ICMP	98 Echo (ping) request	id=0x0017, seq=50/12800, ttl=64 (reply in 157)
157	38.080091	10.0.2.20	10.0.0.20	ICMP	98 Echo (ping) reply	id=0x0017, seq=50/12800, ttl=62 (request in 156)
158	38.912003	10.0.0.20	10.0.10.1	ICMP	98 Echo (ping) request	id=0x000f, seq=1647/28422, ttl=64 (reply in 159)
159	38.912026	10.0.10.1	10.0.0.20	ICMP	98 Echo (ping) reply	id=0x000f, seq=1647/28422, ttl=64 (request in 158)
160	39.103971	10.0.0.20	10.0.2.20	ICMP	98 Echo (ping) request	id=0x0017, seq=51/13056, ttl=64 (no response found!)
161	39.936983	10.0.0.20	10.0.10.1	ICMP	98 Echo (ping) request	id=0x000f, seq=1648/28678, ttl=64 (reply in 162)
162	39.937012	10.0.10.1	10.0.0.20	ICMP	98 Echo (ping) reply	id=0x000f, seq=1648/28678, ttl=64 (request in 161)
163	40.128963	10.0.0.20	10.0.2.20	ICMP	98 Echo (ping) request	id=0x0017, seq=52/13312, ttl=64 (no response found!)
164	40.960970	10.0.0.20	10.0.10.1	ICMP	98 Echo (ping) request	id=0x000f, seq=1649/28934, ttl=64 (reply in 165)

A queda do enlace **R0-R2** interrompeu o tráfego para o **pc2**, pois as rotas estáticas configuradas não incluíam caminhos alternativos.

PARTE 2 - Testando campo TTL com loop na rede

6. Gere um tráfego único a partir do pc0 para o pc2: ping -c1 10.0.2.20

```
root@pc0:/# ping -c1 10.0.2.20
PING 10.0.2.20 (10.0.2.20) 56(84) bytes of data.
3 From 10.0.0.1 icmp_seq=1 Time to live exceeded

--- 10.0.2.20 ping statistics ---
1 packets transmitted, 0 received, +1 errors, 100% packet loss, time 0ms

root@pc0:/#
```

1. Qual mensagem de erro foi recebida no terminal do pc0?

From 10.0.0.1 icmp_seq=1 Time to live exceeded

Analizando as capturas dos Wireshark responda:

1. Aproximadamente em qual roteador o pacote foi descartado? Procure pelo menor valor de ttl.

No R2 em eth1 o menor ttl registrado é 1, então o pacote deve ter sido descartado nele.

*- [eth2@R1 (ib932)]									
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help									
Apply a display filter ... <Ctrl-/>									
No.	Time	Source	Destination	Protocol	Length	Info			
1	0.000000	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=62 (no response found!)			
2	0.000042	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=59 (no response found!)			
3	0.000072	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=56 (no response found!)			
4	0.000098	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=53 (no response found!)			
5	0.000117	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=50 (no response found!)			
6	0.000133	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=47 (no response found!)			
7	0.000150	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=44 (no response found!)			
8	0.000170	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=41 (no response found!)			
9	0.000194	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=38 (no response found!)			
10	0.000217	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=35 (no response found!)			
11	0.000245	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=32 (no response found!)			
12	0.000262	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=29 (no response found!)			
13	0.000279	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=26 (no response found!)			
14	0.000295	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=23 (no response found!)			
15	0.000311	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=20 (no response found!)			
16	0.000328	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=17 (no response found!)			
17	0.000344	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=14 (no response found!)			
18	0.000360	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=11 (no response found!)			
19	0.000376	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=8 (no response found!)			
20	0.000393	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=5 (no response found!)			
21	0.000409	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=2 (no response found!)			
22	5.249055	42:00:aa:00:00:0b	42:00:aa:00:00:0a	ARP	42	Who has 10.0.12.2? Tell 10.0.12.1			

*-[eth1@R0 (ib932)]

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	fe80::108c:92ff:fe1...	ff02::2	ICMPv6	70	Router Solicitation from 2a:9a:66:ab:ea:5e
2	63.871971	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=63 (no response found!)
3	63.872028	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=60 (no response found!)
4	63.872057	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=57 (no response found!)
5	63.872084	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=54 (no response found!)
6	63.872106	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=51 (no response found!)
7	63.872123	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=48 (no response found!)
8	63.872139	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=45 (no response found!)
9	63.872157	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=42 (no response found!)
10	63.872181	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=39 (no response found!)
11	63.872204	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=36 (no response found!)
12	63.872230	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=33 (no response found!)
13	63.872252	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=30 (no response found!)
14	63.872268	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=27 (no response found!)
15	63.872285	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=24 (no response found!)
16	63.872301	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=21 (no response found!)
17	63.872317	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=18 (no response found!)
18	63.872333	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=15 (no response found!)
19	63.872350	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=12 (no response found!)
20	63.872366	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=9 (no response found!)
21	63.872382	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=6 (no response found!)
22	63.872398	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=3 (no response found!)

*-[eth1@R2 (ib932)]

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=61 (no response found!)
2	0.000033	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=58 (no response found!)
3	0.000061	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=55 (no response found!)
4	0.000087	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=52 (no response found!)
5	0.000104	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=49 (no response found!)
6	0.000120	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=46 (no response found!)
7	0.000136	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=43 (no response found!)
8	0.000159	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=40 (no response found!)
9	0.000182	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=37 (no response found!)
10	0.000208	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=34 (no response found!)
11	0.000233	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=31 (no response found!)
12	0.000249	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=28 (no response found!)
13	0.000265	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=25 (no response found!)
14	0.000282	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=22 (no response found!)
15	0.000298	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=19 (no response found!)
16	0.000314	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=16 (no response found!)
17	0.000330	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=13 (no response found!)
18	0.000347	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=10 (no response found!)
19	0.000363	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=7 (no response found!)
20	0.000379	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=4 (no response found!)
21	0.000396	10.0.0.20	10.0.2.20	ICMP	98	Echo (ping) request id=0x0019, seq=1/256, ttl=1 (no response found!)

2. Qual o significado da linha, apresentada no terminal do pc0, com o seguinte conteúdo parcial: Time-to-live exceeded (Time to live exceeded in transit)?

Quando o TTL chega a 0, o pacote não pode mais ser roteado. Isso é um mecanismo de segurança para evitar loops infinitos na rede.

3. Quando o TTL chega a 0, o pacote não pode mais ser roteado. Isso é um mecanismo de segurança para evitar loops infinitos na rede.

Evitar loops infinitos.