

Nome: Nathan Medeiros Cristiano.

Turma: **RED129005**

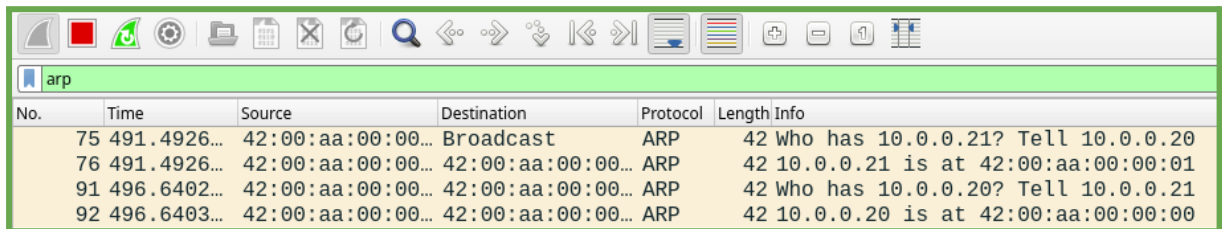
**LABORATÓRIO 12**

**USANDO MINHA MÁQUINA / IFSC**

**ARP, ND e IPv6**

## 12.2 ARP

- Faça um ping do PC1 no PC2 e observe a sequência de pacotes trocados.



The image shows a Wireshark packet capture window with the filter 'arp'. It displays four ARP-related packets:

No.	Time	Source	Destination	Protocol	Length	Info
75	491.4926...	42:00:aa:00:00...	Broadcast	ARP	42	Who has 10.0.0.21? Tell 10.0.0.20
76	491.4926...	42:00:aa:00:00...	42:00:aa:00:00...	ARP	42	10.0.0.21 is at 42:00:aa:00:00:01
91	496.6402...	42:00:aa:00:00...	42:00:aa:00:00...	ARP	42	Who has 10.0.0.20? Tell 10.0.0.21
92	496.6403...	42:00:aa:00:00...	42:00:aa:00:00...	ARP	42	10.0.0.20 is at 42:00:aa:00:00:00

- Anotando o Cache deste primeiro ping

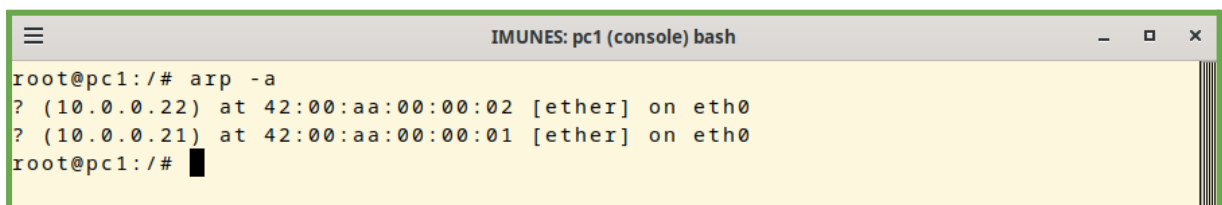


The image shows four terminal windows, each titled 'IMUNES: pcX (console) bash', displaying the output of the 'arp -a' command:

- pc1:** root@pc1:/# arp -a  
? (10.0.0.21) at 42:00:aa:00:00:01 [ether] on eth0
- pc2:** root@pc2:/# arp -a  
? (10.0.0.20) at 42:00:aa:00:00:00 [ether] on eth0
- pc4:** root@pc4:/# arp -a
- pc3:** root@pc3:/# arp -a

- Faça um ping do PC1 para o PC3.

As tabelas foram atualizadas após o ping?



The image shows a terminal window titled 'IMUNES: pc1 (console) bash' with the output of 'arp -a' after a ping from PC1 to PC3:

```
root@pc1:/# arp -a
? (10.0.0.22) at 42:00:aa:00:00:02 [ether] on eth0
? (10.0.0.21) at 42:00:aa:00:00:01 [ether] on eth0
root@pc1:/#
```

Sim, foi atualizado!

- Faça um ping do PC1 para o PC4.

As tabelas foram atualizadas após o ping?

```

IMUNES: pc1 (console) bash
root@pc1:/# arp -a
? (10.0.0.22) at 42:00:aa:00:00:02 [ether] on eth0
? (10.0.0.21) at 42:00:aa:00:00:01 [ether] on eth0
? (10.0.0.23) at 42:00:aa:00:00:03 [ether] on eth0
root@pc1:/# █

```

Sim, foi atualizado!

- Confira, anote e explique o significado do cache do arp da sua máquina real (não Imunes) fazendo:

```

Arquivo  Editar  Ver  Pesquisar  Terminal  Ajuda
aluno: ~$ ip neigh show
191.36.13.126 dev eth0 lladdr ac:71:2e:08:c8:e8 REACHABLE
191.36.13.77 dev eth0 lladdr a8:a1:59:95:c9:1f REACHABLE
fe80::ae71:2eff:fe08:c8e8 dev eth0 lladdr ac:71:2e:08:c8:e8 router REACHABLE
aluno: ~$ █

```

Aqui aparecem dois IPS, em suas duas versões IPV4 e IPV6, juntamente com o MAC de cada dispositivo.

## 12.3 ND e IPv6

- *Neighbor Discovery*

```

root@pc2:/# ndisc6 -m fc00:2::1 eth0
Soliciting fc00:2::1 (fc00:2::1) on eth0...
Target link-layer address: 42:00:AA:00:00:06
from fc00:2::1
root@pc2:/# █

```

- Faça um ping6 entre o pc3 ao pc2:

```

root@pc3:/# ip addr add fc00:2::21/64 dev eth0
root@pc3:/# ping6 -c1 fc00:2::20
PING fc00:2::20(fc00:2::20) 56 data bytes
64 bytes from fc00:2::20: icmp_seq=1 ttl=64 time=0.261 ms

--- fc00:2::20 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 0.261/0.261/0.261/0.000 ms
root@pc3:/# █

```

- Se tudo estiver devidamente configurado, deve-se obter sucesso no ping entre o pc3 e pc2. Entrega direta ou indireta?

Obteve sucesso! foi entrega direta

- Faça um ping6 entre o pc3 ao pc0.

```
root@pc3:/# ping6 -c1 fc00::20
ping6: connect: Network is unreachable
root@pc3:/# █
```

Não obteve sucesso! pois o pc0 está em uma sub-rede diferente de pc3, por conta disso é necessário configurar um gateway padrão ao pc3, para fazer a entrega indireta ao pc0.

- Configurando PC3

Adotando um "Default Gateway"

```
root@pc3:/# ip addr show dev eth0

2: eth0@if11: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP
group default qlen 50
    link/ether 42:00:aa:00:00:05 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet6 fc00:2::21/64 scope global
        valid_lft forever preferred_lft forever
    inet6 fe80::4000:aaff:fe00:5/64 scope link
        valid_lft forever preferred_lft forever
root@pc3:/# █
```

```
root@pc3:/# ip -6 route show
::1 dev lo0 proto kernel metric 256 pref medium
fc00:2::/64 dev eth0 proto kernel metric 256 pref medium
fe80::/64 dev eth0 proto kernel metric 256 pref medium
root@pc3:/# █
```

```
root@pc3:/# ip -6 route add default via fc00:2::1 dev eth0
root@pc3:/# ip -6 route show
::1 dev lo0 proto kernel metric 256 pref medium
fc00:2::/64 dev eth0 proto kernel metric 256 pref medium
fe80::/64 dev eth0 proto kernel metric 256 pref medium
default via fc00:2::1 dev eth0 metric 1024 pref medium
```

- **Ping entre pc3 e pc0 novamente!**

Obteve sucesso? Sim ou não e por quê?

```
root@pc3:/# ping6 -c4 fc00::20
PING fc00::20(fc00::20) 56 data bytes
64 bytes from fc00::20: icmp_seq=1 ttl=62 time=0.272 ms
64 bytes from fc00::20: icmp_seq=2 ttl=62 time=0.148 ms
64 bytes from fc00::20: icmp_seq=3 ttl=62 time=0.163 ms
64 bytes from fc00::20: icmp_seq=4 ttl=62 time=0.148 ms
```

Sim, foi obtido sucesso entre os testes. Funcionou agora por conta da configuração feita, foi adotado um "Default Gateway" para o pc3.

- **Traceroute6 e anote a rota para todos os demais PCs**

Baseado na captura de pacotes do Wireshark explique o processo de descoberta de vizinhança (*Neighbor Solicitation* - NS e *Neighbor Advertisement* - NA), citando os endereços de multicast e link local utilizados. Obs.: ao final do roteiro há alguns exemplos de mensagens.

Numa mensagem do tipo *Neighbor Solicitation* qual é o endereço IPv6 de origem e destino? Explique/defina ambos.

```
root@pc0:/# traceroute6 fc00:1::20
traceroute to fc00:1::20 (fc00:1::20) from fc00::20, port 33434, from port 65278
, 30 hops max, 60 bytes packets
 1 fc00::1 (fc00::1) 0.003 ms 0.157 ms 0.030 ms
 2 fc00:1::20 (fc00:1::20) 0.013 ms 0.021 ms 0.002 ms
root@pc0:/# traceroute6 fc00:2::20
traceroute to fc00:2::20 (fc00:2::20) from fc00::20, port 33434, from port 65277
, 30 hops max, 60 bytes packets
 1 fc00::1 (fc00::1) 0.003 ms 0.075 ms 0.184 ms
 2 fc00:3::1 (fc00:3::1) 0.027 ms 0.145 ms 0.087 ms
 3 fc00:2::20 (fc00:2::20) 0.042 ms 0.034 ms 0.002 ms
root@pc0:/# traceroute6 fc00:2::21
traceroute to fc00:2::21 (fc00:2::21) from fc00::20, port 33434, from port 65276
, 30 hops max, 60 bytes packets
 1 fc00::1 (fc00::1) 0.003 ms 0.080 ms 0.212 ms
 2 fc00:3::1 (fc00:3::1) 0.027 ms 0.195 ms 0.070 ms
 3 fc00:2::21 (fc00:2::21) 0.044 ms 0.026 ms 0.002 ms
root@pc0:/#
```

Time	Source	Destination	Protocol	Length	Info
116 1952.769979	fe80::4000:aaff:fe0...	fc00:2::20	ICMPv6	86	Neighbor Solicitation for fc00:2::20 from 42:00:aa:00:00:06
117 1952.770108	fe80::4000:aaff:fe0...	fc00:2::1	ICMPv6	86	Neighbor Solicitation for fc00:2::1 from 42:00:aa:00:00:04
118 1952.770121	fc00:2::1	fe80::4000:aaff:fe0...	ICMPv6	78	Neighbor Advertisement fc00:2::1 (rtr, sol)
119 1952.770152	fc00:2::20	fe80::4000:aaff:fe0...	ICMPv6	78	Neighbor Advertisement fc00:2::20 (sol)
120 1957.889740	fe80::4000:aaff:fe0...	fe80::4000:aaff:fe0...	ICMPv6	86	Neighbor Solicitation for fe80::4000:aaff:fe00:4 from 42:00:aa:00:00:06
121 1957.889813	fe80::4000:aaff:fe0...	fe80::4000:aaff:fe0...	ICMPv6	86	Neighbor Solicitation for fe80::4000:aaff:fe00:6 from 42:00:aa:00:00:04
122 1957.889838	fe80::4000:aaff:fe0...	fe80::4000:aaff:fe0...	ICMPv6	78	Neighbor Advertisement fe80::4000:aaff:fe00:6 (rtr, sol)
123 1957.889859	fe80::4000:aaff:fe0...	fe80::4000:aaff:fe0...	ICMPv6	78	Neighbor Advertisement fe80::4000:aaff:fe00:4 (sol)
124 1961.726010	fe80::4000:aaff:fe0...	ff02::9	RIPng	146	Command Response, Version 1
125 1963.790298	fc00:2::20	fc00:2::21	UDP	74	65276 → 33437 Len=12
126 1963.790407	fc00:2::21	fc00:2::20	ICMPv6	122	Destination Unreachable (Port unreachable)
127 1963.791009	fc00:2::20	fc00:2::21	UDP	74	65276 → 33437 Len=12
128 1963.791037	fc00:2::21	fc00:2::20	ICMPv6	122	Destination Unreachable (Port unreachable)
129 1963.791348	fc00:2::20	fc00:2::21	UDP	74	65276 → 33437 Len=12
130 1963.791380	fc00:2::21	fc00:2::20	ICMPv6	122	Destination Unreachable (Port unreachable)
131 1968.897963	fe80::4000:aaff:fe0...	fc00:2::21	ICMPv6	86	Neighbor Solicitation for fc00:2::21 from 42:00:aa:00:00:06
132 1968.898072	fe80::4000:aaff:fe0...	fc00:2::1	ICMPv6	86	Neighbor Solicitation for fc00:2::1 from 42:00:aa:00:00:05
133 1968.898086	fc00:2::1	fe80::4000:aaff:fe0...	ICMPv6	78	Neighbor Advertisement fc00:2::1 (rtr, sol)
134 1968.898115	fc00:2::21	fe80::4000:aaff:fe0...	ICMPv6	78	Neighbor Advertisement fc00:2::21 (sol)

O "Neighbor Solicitation - NS e Neighbor Advertisement - NA" tem um princípio de funcionamento equivalente ao protocolo ARP do IPV4, usado para descobrir o endereço de MAC do dispositivo vinculado ao respectivo IP.

- **Funcionamento:** O computador (N) quer enviar um pacote para o (X), mas só sabe o seu endereço IPv6.

O (N)envia uma mensagem **NS** (Solicitação) para um endereço de multicast especial na rede local.

A mensagem seria semelhante a dizer "Olá a todos na rede, **quem de vocês tem este endereço IPv6?**" (Não é igual ao Broadcast).

O (X), que tem aquele endereço IPv6, recebe a mensagem de solicitação.

Ele responde com uma mensagem **NA (Anúncio)** enviada *diretamente* de volta ao N.

A mensagem é: "**Eu tenho esse IP!** O meu endereço MAC (físico) é este aqui."

116	2025-11-15 18:00...	fe80::4000:aaf...	fc00:2::20	ICMP...	86 Neighbor Solicitation for fc00:2::20 from 42:00:aa:00:00:06
117	2025-11-15 18:00...	fe80::4000:aaf...	fc00:2::1	ICMP...	86 Neighbor Solicitation for fc00:2::1 from 42:00:aa:00:00:04
118	2025-11-15 18:00...	fc00:2::1	fe80::4000:aaf...	ICMP...	78 Neighbor Advertisement fc00:2::1 (rtr, sol)
119	2025-11-15 18:00...	fc00:2::20	fe80::4000:aaf...	ICMP...	78 Neighbor Advertisement fc00:2::20 (sol)

- **Entendendo o "ip -6 neighbor show"**

**Qual é a funcionalidade desse comando? :**

Ele exibe a tabela de vizinhança no cache do respectivo dispositivo. Sendo equivalente ao "arp-n" do IPV4.

**Qual é o significado do conteúdo dessa tabela? :**

Ele detalha o IPV6 do vizinho, a interface utilizada na comunicação entre os dispositivos, e o endereço MAC do vizinho.

**A tabela mostrada em cada um dos casos é compatível com o diagrama da rede montado? :**

Sim é compatível. A imagem do comando vai mostrar! que cada dispositivo só "enxerga" sua vizinhança imediata, (sub-rede).

Por que, por exemplo, na tabela do pc2 não há uma referência explícita ao pc0? :

Porque pc0 e pc2 não são vizinhos.

Explique sucintamente as diferenças na comunicação e na montagem da tabela IP x MAC, em IPv4 e IPv6. :

#### O IPV4:

Para descobrir um MAC, um PC envia um **ARP Request** (Solicitação) para o endereço de broadcast (um "grito" para todos na rede).

#### o IPV6:

Para descobrir um MAC, um PC envia uma **Neighbor Solicitation** (Solicitação) para um endereço de multicast especial (muito mais eficiente, pois não "grita" para todos, apenas para os interessados). A resposta é um **Neighbor Advertisement** (Anúncio).



```
IMUNES: pc0 (console) bash
root@pc0:/# ip -6 neighbor show
fe80::4000:aaff:fe00:0 dev eth0 lladdr 42:00:aa:00:00:00 router STALE
fc00::1 dev eth0 lladdr 42:00:aa:00:00:00 router STALE
root@pc0:/#

IMUNES: pc1 (console) bash
root@pc1:/# ip -6 neighbor show
fc00:1::1 dev eth0 lladdr 42:00:aa:00:00:02 router STALE
fe80::4000:aaff:fe00:2 dev eth0 lladdr 42:00:aa:00:00:02 router STALE
root@pc1:/#

IMUNES: pc2 (console) bash
root@pc2:/# ip -6 neighbor show
fc00:2::1 dev eth0 lladdr 42:00:aa:00:00:06 router STALE
fe80::4000:aaff:fe00:6 dev eth0 lladdr 42:00:aa:00:00:06 router STALE
fe80::4000:aaff:fe00:5 dev eth0 lladdr 42:00:aa:00:00:05 STALE
fc00:2::21 dev eth0 lladdr 42:00:aa:00:00:05 STALE
root@pc2:/#

IMUNES: pc3 (console) bash
root@pc3:/# ip -6 neighbor show
fc00:2::1 dev eth0 lladdr 42:00:aa:00:00:06 router STALE
fe80::4000:aaff:fe00:6 dev eth0 lladdr 42:00:aa:00:00:06 router STALE
fe80::4000:aaff:fe00:4 dev eth0 lladdr 42:00:aa:00:00:04 STALE
fc00:2::20 dev eth0 lladdr 42:00:aa:00:00:04 STALE
root@pc3:/#
```