Protocolo IPV6

Conceitos Gerais

1. Introdução

- O protocolo IPv6 surgiu como evolução e solução para a escassez de endereços IPv4. Apareceu como solução definitiva para tal problema, e não mais paliativa como o NAT.

- Atualmente, alguns países e continentes estavam passando por uma severa falta de endereços IP's, o que compromete a evolução e funcionamento de determinadas tecnologias.

- Uma das áreas mais afetadas é a automação industrial e a Internet das Coisas (IOT), que faz uso de um grande número de IP para conectar os diversos equipamentos.

- Em termos gerais, o IPV6 é muito parecido com o IPV4, no geral, mudam apenas a capacidade e algumas características específicas.

2. IOT

- A **IoT (Internet das Coisas)** é a conexão de **objetos físicos à internet**, permitindo que eles coletem dados, se comuniquem e tomem decisões automáticas. Em resumo, tornar objetos comuns em objetos inteligentes e conectados.

- Exemplos do cotidiano: Lâmpada inteligente, Smart TV, Relógio inteligente e Geladeira conectada.

- A arquitetura de um sistema IoT geralmente tem

1. Sensores → coletam dados (temperatura, movimento, etc.)

2. Conectividade → Wi-Fi, 4G/5G, Bluetooth

3. Processamento → servidor ou nuvem

4. Ação → resposta automática

2. IOT

- Geralmente, os sistemas IOT usam computação em nuvem, IA, e protocolos sem fio de comunicação. As principais características de um sistema IOT:

1. Conectividade

- Dispositivos conectados à internet

2. Automação

- Executam ações sem intervenção humana

3. Coleta de dados

- Monitoramento em tempo real

4. Escalabilidade

- Milhares/milhões de dispositivos

5. Integração

- Comunicação entre diferentes sistemas

6. Baixo consumo de energia

- Dispositivos eficientes (ex: sensores)

2. IOT

- Alguns exemplos de utilização de IOT:

- A TESLA possui carros conectados à internet, atualizações remotas (OTA) e Monitoramento em tempo real.

- A AMAZON possui a Alexa (assistente inteligente) e Casas inteligentes (smart home)

- A Siemens monitora suas máquinas para controlar produção e fazer manutenção preditiva

2. IOT

- Áreas em que a IOT é usada:

🏠 Casa inteligente

- Automação residencial

🏥 Saúde

- Monitoramento de pacientes

🚦 Cidades inteligentes

- Semáforos inteligentes
- Monitoramento de trânsito

🏭 Indústria

- Máquinas conectadas

🌾 Agricultura

- Sensores de solo e clima

2. IOT

- Apesar de muitas vantagens, o IOT ainda possui muitos problemas e desafios, tais como:

1 - Segurança: Invasão

2 – Privacidade: IoT coleta MUITOS dados, e pode comprometer a privacidade

3 - Dependência da internet: Sem internet, para o serviço

4 – Consumo de energia

5 – Falta de padronização: Muitos equipamento e protocolos ainda são incompatíveis entre si.

6 – Escalabilidade: Conforme cresce a rede, o gerenciamento é mais caro e complexo.

2. IOT

- Já houveram casos reais envolvendo segurança em IOT:

- Em 2015, carros da marca JEEP foram invadidos, possibilitando o controle dos freios e outros serviços do carro. Todos os carros foram para o recall.

- Já houveram casos de invasão em dispositivos de hospitais. Como resultado, cirurgias canceladas e muito prejuízo no atendimento das pessoas.

- Já invadiram e roubaram o banco de dados de um cassino nos EUA por meio de um termometro inteligente de um aquário.

- A HONDA em 2020 teve seus equipamentos invadidos, comprometendo a produção global da empresa.

2. IOT

- Já invadiram as câmeras corporais de policiais dos EUA, comprometendo investigações.

- Invadiram equipamentos do Porto de Antuérpia com o objetivo de controlar cargas e facilitar o tráfico de drogas.

- **Muitas vezes, atacando apenas um dispositivo da rede (como uma câmera), toda a rede pode ser comprometida.**

3. Protocolo IPV6

- O protocolo IPV6 utiliza endereços de 128 bits, o que permite um conjunto enorme de possibilidades de endereçamento.

- O IPV6 permite aplicar o princípio surgido pelo IPV4 de endereçar de forma pública e visível na Internet qualquer dispositivo na rede, característica fundamental para o IOT.

- Esses endereços são escritos na forma hexadecimal, diferentemente do IPv4, que utilizava o formato decimal.

- Cada bloco de 4 bits é agrupado em um número hexadecimal e estes números hexadecimais são agrupados a cada 4 dígitos que variam de 0000 a FFFF, gerando 8 grupos. Como exemplo de um endereço IPV6: 2001:0DB8:00AD:000F:3456:AF42:CDCC:0001

3. Protocolo IPV6

- Para se ter uma ideia, o IPV4 (tamanho de 32 bits) possibilita um total de endereços de 2^32, que é aproximadamente **4,3 bilhões.** Já o IPV6 possui 128 bits, com possibilidade de 340 undecilhões de endereços. São bilhões de endereços para cada pessoa da terra.

- Como observado no exemplo, lembrar de endereços IPv4 já não era tarefa das mais fáceis, mas os endereços IPV6 tornam essa tarefa impossível.

- Por essa razão, protocolos auxiliares como DNS e DHCP são fundamentais para facilitar o uso do IPV6.

- Para facilitar a escrita dos endereços IPV6, algumas técnicas de redução de endereços podem ser implementadas, como a supressão de 0's consecutivos, por exemplo: O endereço 2001.CAFE:04FF:0000:0000:0000:0000:00CC pode ser escrito 2001:CAFE:4FF::CC.

3. Protocolo IPV6

- Uma característica do endereçamento IPv6 é, devido ao grande volume de possibilidades de endereços, definiu-se uma máscara padrão de distribuição para LAN's de endereços /64, ou seja, são 64 bits para rede e 64 bits para hosts.

- Assim, um usuário que contrata o serviço de uma operadora tem endereços suficientes para estruturar todos os seus objetos que se conectam à Internet.

- Outro ponto importante é a capacidade de autoconfiguração dos hosts frente ao IPv6, não dependendo do protocolo DHCP para conectividade, ainda que ele exista o chamado de DHCPv6.

- Os hosts conseguem trocar informações entre si e com os roteadores da rede para realizar a configuração automática dos parâmetros de rede. Por isso, o IPV6 é um protocolo **plug-and-play**. Essas interfaces são chamadas de LINK-LOCAL, provendo o mínimo de comunicação na rede local.

3. Protocolo IPV6

- Geralmente, quando se contrata um provedor que utiliza o IPV6, ele fornece um bloco **/56** ou **/48.** Isso permite que seja criado várias redes /64 dentro da sua casa ou empresa.

- Por exemplo, imagine que o provedor forneceu `2001:db8:abcd:1200::/56.` Isso significa que os **primeiros 56 bits são fixos.** O resto pode ser usado livremente. Cada rede local no IPv6 usa **/64, sendo possível** dividir /56 em várias redes /64. Por exemplo:

```
2001:db8:abcd:1200::/56
```

Você pode criar:

Rede	Prefixo
Wi-Fi principal	2001:db8:abcd:1200::/64
IoT (câmeras, Alexa)	2001:db8:abcd:1201::/64
Visitantes	2001:db8:abcd:1202::/64
Trabalho	2001:db8:abcd:1203::/64

3. Protocolo IPV6

- Uma dúvida muito comum é se uma rede é /56, deveriam restar 72 bits para hosts. No IPv6 existe uma **regra importante**: **Os hosts sempre usam 64 bits (interface ID).**

- O endereço é dividido em: **8 bits → para criar sub-redes e 64 bits → para hosts (obrigatório). O** IPv6 foi projetado para que t**oda rede local seja /64.** Isso é essencial para SLAAC(autoconfiguração), correto funcionamento do protocolo e compatibilidade entre hosts.

Parte	Bits	Função
Prefixo do provedor	56	fixo
Sub-rede	8	você escolhe
Host (interface ID)	64	dispositivos

4. Cabeçalho IPV6

- O cabeçalho IPv6 tem como característica o fato de ser mais simples e modular, quando comparado ao IPv4.

- O termo mais simples é no sentido de possuir uma quantidade menor de campos e não ser variável, tendo tamanho fixo de 40 bytes, diferentemente do IPv4 (varia de 20 a 60 bytes).

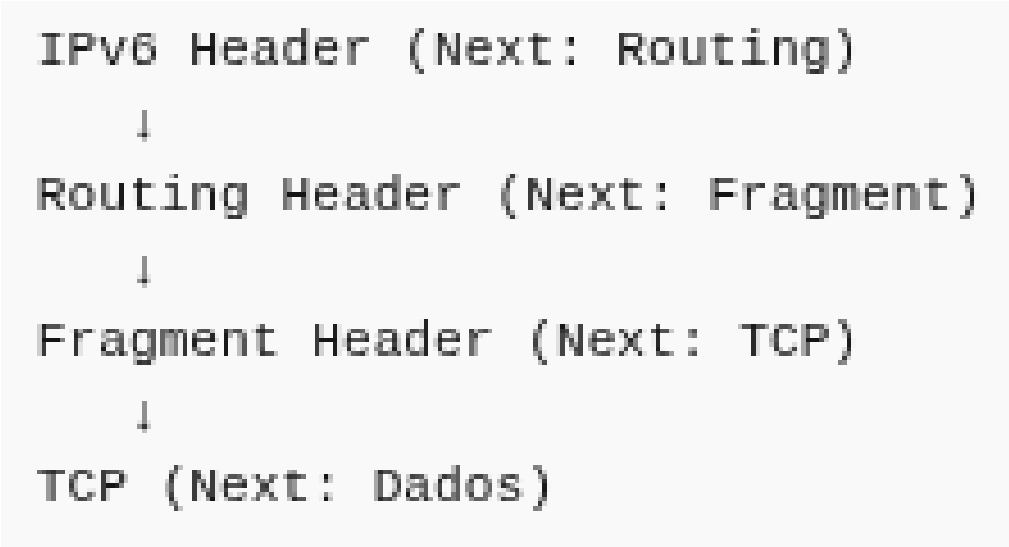
- Além disso, diz-se que é **modular** por permitir a utilização de cabeçalhos de extensão através da utilização de ponteiros entre os cabeçalhos para o provimento de outros serviços.

- O termo Modular significa **dividido em partes independentes (módulos).** No IPv6 o cabeçalho principal é mínimo (40 bytes) e as funcionalidades extras são adicionadas **separadamente**, em blocos chamados **Cabeçalhos de extensão (extension headers).**

4. Cabeçalho IPV6

- Um pacote IPv6 não é só um cabeçalho, ele vira uma **cadeia (chain).** Cada parte tem um campo chamado **Next Header,** que funciona como um "ponteiro" para o próximo bloco, criando uma sequência encadeada.

- A imagem abaixo mostra um exemplo de como o IPV6 é modular, mostrando uma sequencia:



4. Cabeçalho IPV6

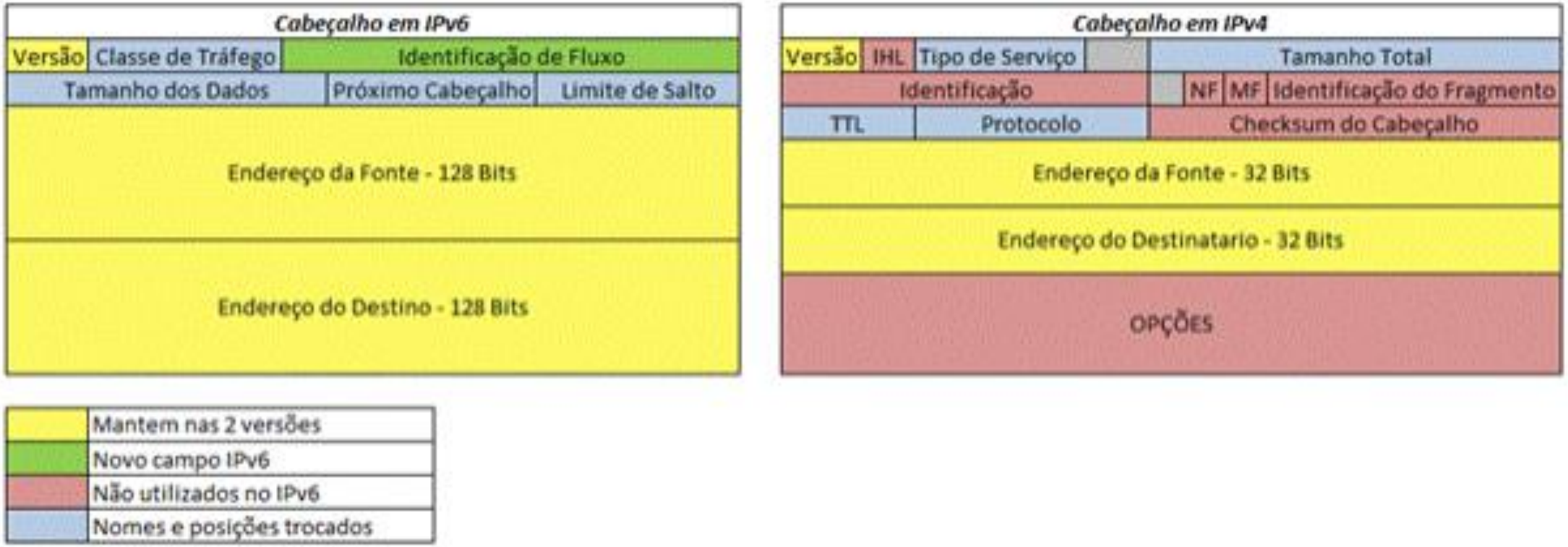
- No cabeçalho IPV6 adiciona apenas o que precisa, onde cada funcionalidade é independente, não poluindo o cabeçalho principal.

- O IPV4 não é modular, deixando o cabeçalho mais poluído. No IPV6 o cabeçalho principal tem apenas o básico, os extras são separados, deixando muito mais organizado.

- Com isso, o IPV6 tem mais performance, pois os roteadores conseguem ler o pacote com mais agilidade.

4. Cabeçalho IPV6

- A imagem abaixo mostra um comparativo entre os cabeçalhos IPV4 e IPV6:



4. Cabeçalho IPV6

- Analisando os campos do cabeçalho IPV6:

- **Versão**: Campo de 4 bits indicando a versão 6 do protocolo.

- **Classe de Tráfego**: Campo de 8 bits. Possui função similar à função de ToS do IPv4 na classificação e priorização de tráfego.

- **Identificação de Fluxo**: Campo de 20 bits utilizado para marcação do tráfego. Identifica um fluxo, que basicamente é um conjunto de pacotes que fazem parte da **mesma comunicação**.

- **Tamanho do campo de dados**: Campo de 16 bits, indicando o tamanho APENAS dos dados enviados junto ao cabeçalho.

4. Cabeçalho IPV6

- **Próximo Cabeçalho:** Campo de 8 bits. Campo utilizado para referenciar os cabeçalhos de extensão eventualmente utilizados. Caso não haja cabeçalho de extensão, será constada a informação do protocolo de camada superior, tal qual consta no cabeçalho IPv4.

- **Limite de Salto**: Campo de 8 bits. Campo similar ao TTL do IPv4, determinando o tempo de vida ou limite de saltos do pacote na rede.

- **Endereços de Origem e Destino**: Campos utilizados para definir os endereços de 128 bits de origem e destino dos pacotes IPv6.

5. Novidades do IPV6

- O protocolo IPv6 possui implementações de segurança de forma nativa através do suporte ao protocolo IPSec, sendo este obrigatório. No IPv4, este procedimento de segurança é opcional e utilizou os conceitos e técnicas criadas para o IPv6.

- IPSec já foi pensado desde o início, ou seja, o suporte é **obrigatório na implementação** . Todo sistema IPv6 **deve ser capaz de usar IPSec**, mas não quer dizer que ele esteja sempre ativo. Já no IPV4 o IPSec é **opcional,** nem todos os dispositivos implementam.

- Com relação ao tamanho dos pacotes, no IPV4 o tamanho máximo é 64Kb. Isso acontece porque o campo de tamanho tem limite de 16 bits.

- Já no IPV6 o cabeçalho básico **NÃO suporta pacotes maiores que 64 KB diretamente.** Mas existe um recurso chamado **Jumbogram** (pacote maior que 64 KB)

5. Novidades do IPV6

- No IPv6 o campo padrão de tamanho tem limite semelhante ao IPv4. **MAS** existe um recurso chamado **Cabeçalho de extensão,** e um desses cabeçalhos especiais permite informar tamanhos muito maiores (acima de 64 KB).

- Isso pode ser muito útil em Redes de alto desempenho (data centers), Computação científica e Transferência de grandes volumes de dados. Um jumbogram pode chegar até 4GB.

- A fragmentação dos pacotes no IPv6 não ocorre mais nos roteadores intermediários. Caso o pacote maior chegue a algum roteador, ele deve descartar o pacote e enviar uma mensagem ICMPv6 do tipo "packet too big" ao host de origem.

- O host de origem é responsável por ajustar o pacote para encaminhamento.

6. Transição do IPV4 para IPV6

- Infelizmente, não existe um “botão” para desligar o IPv4 e começar o IPV6. Em vez disso, é necessário utilizar **estratégias de transição:**

- **Dual Stack:** Dispositivos rodam IPv4 e IPv6 ao mesmo tempo. Basicamente, se o destino suporta IPv6 → usa IPv6, Se não → usa IPv4. É o método mais comum hoje.

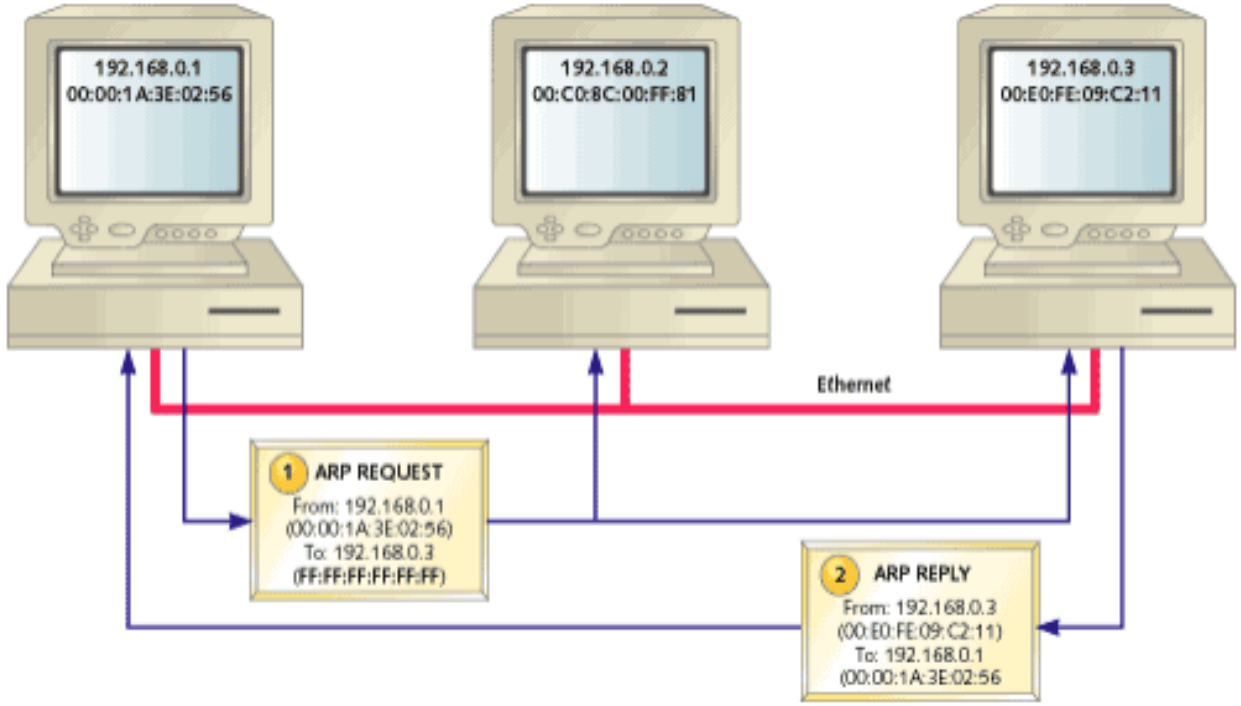
- **Tunelamento ou 6to4**: Nesse modelo, o núcleo da rede opera em IPv4 e as redes de origem e destino operam em IPv6, ou seja, utilizando a infraestrutura atual da internet em IPv4. Dessa forma, os pacotes IPv6 são trafegados dentro de um túnel na rede IPv4.

- **Tradução**: Permite comunicação entre IPv6 → IPv4. Basicamente, um servidor traduz os pacotes entre os dois protocolos. Muito usado por operadoras telefônicas.

7. ARP e RARP

- O **ARP** serve para descobrir o **endereço físico (MAC)** de um dispositivo a partir de um **endereço IP.** Quando um dado é envido em uma rede, o que realmente identifica o destino na camada de enlace é o **MAC**, não o IP.

- A imagem abaixo mostra o modo de operação do ARP:



7. ARP e RARP

- A máquina da esquerda deseja enviar um pacote para o endereço IP de destino 192.168.0.3, porém desconhece o endereço MAC desse dispositivo.

- Nesse caso, ele envia um pacote do tipo "ARP REQUEST" para TODOS os hosts da rede (BROADCAST). Para tanto, utiliza-se como endereço MAC de destino o endereço FF:FF:FF:FF:FF:FF.

- O dispositivo que receber o pacote e for dono do endereço IP de destino em questão, deve responder com um pacote do tipo "ARP REPLY".

- Importante mencionar que, como esse computador recebeu o pacote "ARP REQUEST", ele já foi capaz de mapear o endereço IP e MAC do dispositivo requisitante, que, no caso, é o da esquerda. Portanto, o computador da direita tem condições de emitir uma resposta de forma UNICAST.

7. ARP e RARP

- O dispositivo da esquerda descobre e mapeia em sua tabela ARP a informação adquirida. Da próxima vez, a comunicação entre os mesmos dispositivos será facilitada.

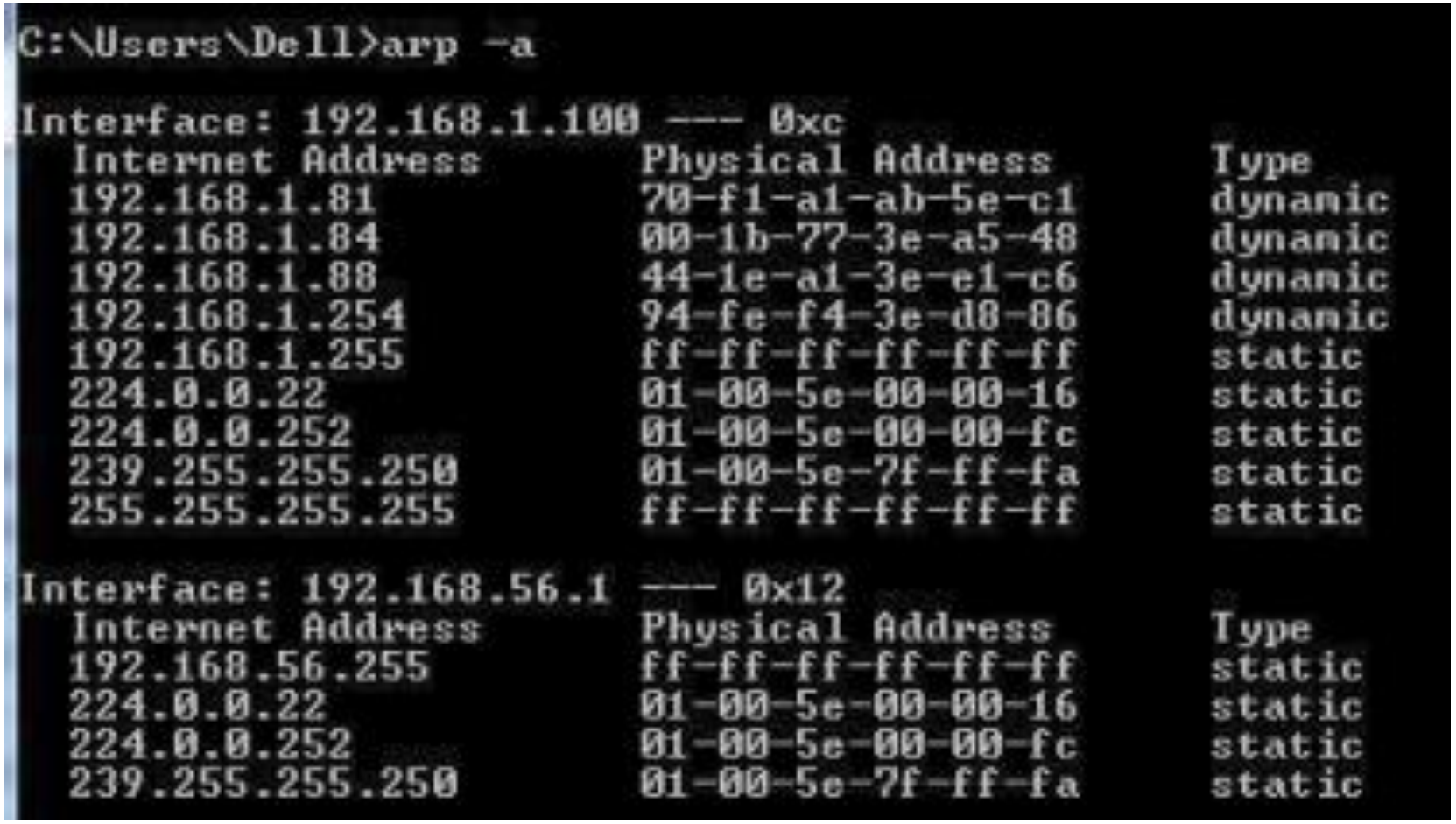
- Os demais dispositivos que receberam via BROADCAST o pacote "ARP REQUEST" não respondem nada, apenas descartam o pacote.

- O **ARP é implementado dentro do sistema operacional**, mais especificamente no seu kernel. Ele faz parte da **pilha de protocolos de rede (TCP/IP),** funciona automaticamente, não precisa instalar nem iniciar manualmente.

- O RARP faz o processo contrário, determina o IP a partir do MAC. Mas está obsoleto, fui substituído por outros serviços como DHCP, DNS, entre outros.

7. ARP e RARP

- Com o comando "arp –a", é possível ver a tabela ARP do dispositivo:



7. ARP e RARP

- Quando se deseja enviar um pacote para um destino fora da rede, também é utilizado o ARP, mas de forma diferente.

- Imagine uma situação onde o PC: 192.168.1.10, com gateway (roteador): 192.168.1.1 deseja enviar um dado para o destino: 8.8.8.8 (fora da rede).

- A ideia consiste em **NÃO usar ARP para o destino final, mas usa ARP para descobrir o MAC do gateway.**

- O PC analisa o IP destino: 8.8.8.8, e conclui que não é da rede. Então ele encaminha o dado para o gateway (roteador).

- O PC precisa descobrir o **MAC do gateway, caso não saiba, utiliza o ARP para descobrir o MAC do** IP 192.168.1.1?" (gateway).

8. ICMP

- O **ICMP** é um protocolo usado para **enviar mensagens de controle e diagnóstico na rede**. Ele **não transporta dados de aplicação,** serve apenas para **informar a situação da rede.**

- **O ICMP serve para responder perguntas como:** “Esse destino existe?”, “O pacote chegou?”, “Teve erro no caminho?” e “A rede está acessível?”. É utilizado muito como protocolo de testes.

- É importante mencionar que o protocolo ICMP e suas mensagens são trafegadas no payload do protocolo IPv4, após o cabeçalho IPv4.

- O ICMP age como se fossem dados para o pacote IP, ele não fica no cabeçalho. Mas, no cabeçalho IP, o campo "Protocolo" será definido como 1 para indicar que a carga útil contém mensagens ICMP

8. ICMP

- Um dos comandos mais famosos do ICMP é o "**ping**", onde basicamente o PC envia um:**ICMP Echo Request** (pedido) e o destinoresponde: **ICMP Echo Reply** (resposta).

- Por exemplo, fazer um PING 192.168.1.3 testa se o dispositivo (com o ip 192.168.1.3) está recebendo pacotes corretamente na rede.

- Outro comando muito utilizado do ICMP é o "traceroute", que mostra todos os passos (roteadores) que o pacote seguiu até o destino, como mostra o exemplo: **traceroute 8.8.8.8**

```
1  192.168.1.1      1 ms   1 ms   1 ms
2  10.0.0.1         5 ms   4 ms   5 ms
3  200.123.x.x     10 ms  11 ms  10 ms
4  8.8.8.8         20 ms  19 ms  21 ms
```

9. IGMP

- O **IGMP (Internet Group Management Protocol)** é um protocolo usado em redes para **gerenciar grupos de multicast**. Em termos simples, ele informa o roteador que deseja receber esse tipo específico de transmissão em grupo.

- Lembrando que existem 3 formas de envio de dados: **Unicast** → 1 para 1 (ex: você acessando um site) , **Broadcast** → 1 para todos na rede e **Multicast** → 1 para vários interessados (mas não todos).

- Desta forma, O IGMP serve para permitir que um host (computador) **entre e saia de um grupo multicast,** bem como informar ao roteador quem quer receber aquele tráfego.

- Esse protocolo é muito usado em IPTV, Streaming ao vivo e Videoconferência.

9. IGMP

- O **IGMP (Internet Group Management Protocol)** é um protocolo usado em redes para **gerenciar grupos de multicast**. Em termos simples, ele informa o roteador que deseja receber esse tipo específico de transmissão em grupo.

- Lembrando que existem 3 formas de envio de dados: **Unicast** → 1 para 1 (ex: você acessando um site) , **Broadcast** → 1 para todos na rede e **Multicast** → 1 para vários interessados (mas não todos).

- Desta forma, O IGMP serve para permitir que um host (computador) **entre e saia de um grupo multicast,** bem como informar ao roteador quem quer receber aquele tráfego.

- Esse protocolo é muito usado em IPTV, Streaming ao vivo e Videoconferência.