* Ler Kurose
* Rede, enlace, físico
* OSI
  + 7 camadas
* ISP (operadoras/provedoras)
  + Conexão
    - N para n
      * Ruim, O(n2)
  + Escalabilidade
  + Elas se conectam entre si nos IXP ou fora dos IXP
  + Exemplo Unisinos no RNP
  + Podem se conectar
  + Privada x pública
* IXP (internet Exchange point)
* Roteadores
  + Podem se comunicar só dentro de ISP ou com IXP
  + Não guarda estado, faz seu melhor no momento
* Backbone
  + é a espinha dorsal da rede mundial, uma infraestrutura de alta capacidade que conecta redes menores e permite a comunicação global
* Algoritmo de roteamento
* Link de grande capacidade
* Sistemas autômatos
* Camada de rede
  + Encaminhar pacotes de origem e destino
  + De forma transparente
    - De equipamento a equipamento até o destino
  + Utilizando diferentes meios de enlace
  + Dar um endereço de origem e destino
  + Endereçamento
    - IP
      * Internet é constituída por sistemas autônomos
      * Best-effort no transporte de datagramas
        + Faz o melhor esforço local, não global
      * Permite a interconexão desses sistemas autônomos
      * Independe do destino, que pode ser uma máquina da subrede local ou de outra subrede
      * Porque IP?
        + Isolamento, modularização
      * Endereços privados e públicos
        + Endereços privados são convertidos para públicos
      * VPN
        + Tudo que passar pela VPN é criptografado, a máquina recebe o mesmo IP da ‘empresa’, o pacote não será aberto, será cifrado, o dado não será decifrado, garante que aquele dado só será acessado por quem tiver o mesmo IP
      * IPv4
        + Cabeçalho

32bits

Normalmente tem tamanho de 20 bytes, porque o campo opções geralmente não é usado

É grande comparado com o UDP que é 8 bytes

Rede com conexão tem cabeçalho menor porque ela guarda o contexto

Rede sem conexão não guarda estado, precisa de mais informações

Serviço com conexão é mais custoso para começar e com conexão é mais custoso durante o caminho, digamos assim

Versão

V4 = &h0100

Best-effort

Ipv4 se recebe ipv6 descarta pacote

IHL

Tamanho do cabeçalho

Serviço

Tipos de serviço que podem ser transportados

Consegue saber a precedência

3 flags

Delay

Throughput

Reliability

A máquina de origem poderia colocar essas opções

Não foi aplicado, não foi efetivo, dependendo da aplicação deveriam ser colocados valores, mas é da máquina, porque dar prioridade para um, e não para outro, é best effort, sem estado, não se consegue implementar para dar preferência para um e não para outro

Alguns protocolos usam esses campos e fazem negociações para usarem

2 bits não usado

Campo opcional para algoritmos de roteamento

Tamanho total

do que tá transportando cabeçalho mais payload

Identificação do pacote

Para identificar fragmentação, conseguir remontar o pacote

Espaço vazio

Para usar para qualquer coisa

Dont frag

Flag

Não tem mais fragmentação

1 bit

More frag

Flag

Tem mais fragmentação

1 bit

Offset

Atua junto com as flags para remontar o pacote

Indica a localização do fragmento

13 bits

Deslocamento para montar

TTL

Tempo de permanência na rede

No final ele é descartado

Resolver problema de loop

É usado com um valor numérico e não com um tempo

Toda vez que passa pelo roteador o número é decrementado

Como definir o valor?

É padronizado

Originalmente 255s

Protocolo

Exemplo TCP

Checksum

Integridade do cabeçalho do payload não

Ninguém viola dados em nível de rede, porque precisa ser rápido o transporte, o perigo é na camada de transporte e aplicação, porque é onde os dados serão processados

Endereço de origem

32 bits

Endereço de destino

32 bits

Opcional

Segurança

Caminho restrito

Indica roteadores obrigatórios

Record route

Timestamp

Engessa a rede

Descontinuado

Tamanho máximo 65535

Tamanho lógico, não físico

Fragmentação

MTU

Unidade de transmissão máxima

O pacote pode ser fragmentado no meio, na metade do caminho, por questões de tecnologia legada

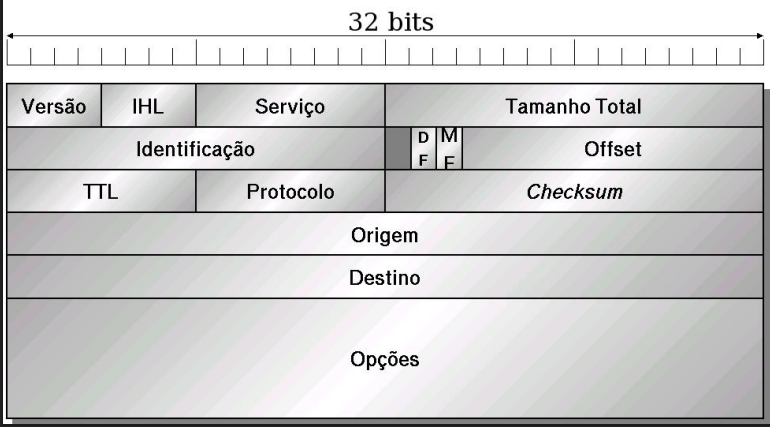
Pode acontecer fragmentação no roteador

Se perder pacote da fragmentação perdeu a camada de transporte que vai tratar

Tem que quebrar a cifra, a VPN cai, porque corrompe, quando montar de novo ele vai ver que foi corrompido, não garante segurança dos dados, para dividir teve que decifrar

Problemas de performance

Para ter melhor proveito de uma VPN ter um MTU baixo, para evitar fragmentação



* + - * + Endereço IP

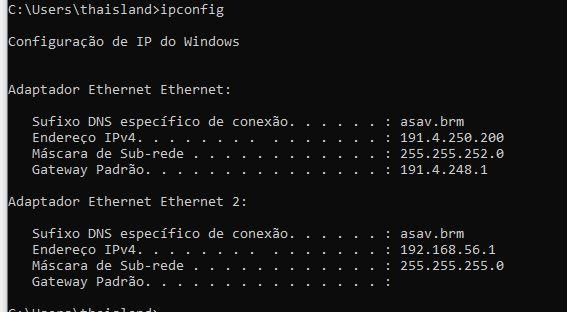
4 octetos

192.168.33.4

Primeira metade é a rede (192.168)

Segunda metade a máquina (33.4)

* + - * + Subrede IP



* + - * + Classes

Endereçamento classful

A

/8

Máscara

255.0.0.0

B

/16

Máscara

255.255.0.0

C

/24

Máscara

255.255.255.0

D

Multicast

E

Reservado pro futuro

* + - * + CIDR

Classless Inter-Domain Routing

* + - * + Endereços especiais

Broadcast

Tudo 1

Loopback

127.

* + - * + DHCP

Dynamic host configuration protocol

Permite que tenha um endereço IP dinamicamente

Pode renovar seu aluguel no endereço em uso

Reutilização de endereços

Atribuição

DHCP discover ->

Chama toda a rede perguntando quem é o DHCP

<- DHCP offer 200.17.98.1

DHCP request 200.17.98.1 ->

<- DHCP 200.17.98.1

DHCP release 200.17.98.1->

Tenta usar o IP antigo

Usava menos o IP público

* + - * + NAT

Network address translation

Dá delay na rede para fazer a conversão

Roteador não implementa NAT por default

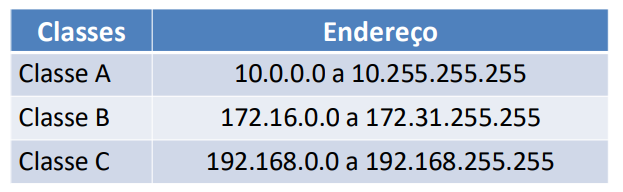
Associa IP e porta

A máquina interna da rede que tem que começar a conexão para que tenha o túnel, se uma máquina fora da rede quiser conectar não consegue diretamente

A tabela NAT é preenchida quando a requisição de dentro da NAT faz a requisição

A tabela NAT é zerada depois de um tempo

* + - * + Endereço IP reservados para redes privadas



* + - * + Tem possibilidade de 2^32 endereços diferentes
        + Idealmente permanecem intactos durante o roteamento, somente os endereços de enlace seriam modificados
        + Descubra qual IP está configurado na sua máquina?

Usar o wireshark e tcpdump

Traceroute destino 2000

* + - * IPv6
  + Roteamento é diferente encaminhamento, o segundo é sobre switch
  + Não garante ordem, garante entrega
  + Não se roteia pacote por pacote, se roteia fluxo
  + Roteadores podem implementar algoritmos diferentes um do outro
  + Roteamento
    - 2 serviços
      * Com conexão
        + Tem uma maior qualidade de serviço
        + Tem origem e destino, todos os pacotes seguem o mesmo fluxo
        + Precisa estabelecer uma conexão antes de enviar os dados
        + Exemplo: redes móveis
        + Dentro da provedora de serviço móvel se conecta com conexão
        + Se querem se conectar através da mesma provedora, um túnel é criado
        + Túnel GTP
        + Circuito virtual
        + Ao se estabelecer com n destinos terá n túneis, não consegue ter milhões de conexões, a capacidade da rede pode ser afetada
        + Se ocorrer um problema em um roteador quebra o túnel, perde a conexão, exemplo guerra
        + Túnel reserva banda nos roteadores, memoria CPU reservada
        + Manter a conexão ou não, depende da implementação da tecnologia implementada no caminho inteiro
        + Escolha do melhor caminho

Algoritmo: menor caminho, latência, etc.

* + - * + A saída de um túnel em uma provedora pode ou não ir para um túnel de outra provedora, mas a provedora garante o túnel dentro do seu domínio apenas
        + Protocolo de sinalização

Inicia, recebe, aceita, chama conexão, fluxo de dados

Conexão lógica por id

* + - * + Pacotes em ordem
        + Pacote com número de CV
        + Setup do circuito obrigatório
        + Espaço nas tabelas do roteador para informação de estado
        + Roteamento é definido na criação
        + Efeito de falhas: fim do circuito virtual onde falhou
        + Qualidade do serviço: durante a criação
      * Sem conexão
        + Caso de uso: Internet, transmissão de datagramas
        + Datagrama
        + Cada pacote pode seguir caminhos diferentes de acordo com a decisão do roteador
        + Pacotes fora de ordem
        + Difícil de garantir a qualidade de serviço
        + Consegue usar melhor o recurso da rede, porque não precisa reservar o recurso, apenas despacha os pacotes de acordo com o algoritmo
        + A rede se mantém mesmo que um nó caia, como em caso de guerra
        + Protocolo

Não tem sinalização

Não tem controle

Envia e recebe diagramas

* + - * + Roteamento é definido por pacotes, independentemente, decisão local
        + Efeitos de falhas: só pacotes perdidos
        + Qualidade do serviço: difícil

Por isso que a camada de transporte criou protocolos para garantir a qualidade

* + Descubra o endereço de IP do seu ISP
    - Docker traceroute google.com.br
    - Print e interpretação do que tá acontecendo
    - Explicação de todas as linhas do tracerouter
* Enlace
  + Encaminhamento