

# IPv6 (Internet Protocol version 6)

## Introdução



**Prof. Luiz Arthur Feitosa dos Santos**

**[luiz.arthur.feitosa.santos@gmail.com](mailto:luiz.arthur.feitosa.santos@gmail.com)/[luizsantos@utfpr.edu.br](mailto:luizsantos@utfpr.edu.br)**

Material licenciado sob [Licença Creative Commons BY 4.0](https://creativecommons.org/licenses/by/4.0/)



# Introdução ao IPv6

O **IPv4**, que serve como base para o funcionamento da Internet vem do final da década de **1970 início de 1980**. Entretanto, com a **popularização dos computadores** e com o surgimento de conceitos como **IoT**, no qual espera-se que cada pessoa/empresa conecte inúmeras dispositivos, os **~4 bilhões endereços do IPv4 basicamente já se esgotaram**.

Para resolver tal problema, **surge na década de 1990 o IPv6 (Internet Protocol version 6)**, que dá a possibilidade de endereçar **1.500 *hosts* por metro quadrado do planeta terra**, com os seus  $2^{128}$  bits de endereço e **resolve o problema da falta de endereços IPs válidos na Internet**.

Então, o **IPv6** é a mais recente versão do protocolo IP, sendo desenvolvido para substituir o IPv4. No entanto, mesmo com esse contexto, **o IPv6 ainda hoje (2025) não substituiu completamente o IPv4**, isso por vários motivos, tais como: medo do IPv6 apresentar problemas desconhecidos. Todavia o IPv6 vem ganhando força para essa substituição nos últimos tempos e por isso merece atenção.

Este texto pressupõe que você já saiba IPv4 em detalhes.

# Motivação para a Criação do IPv6

O esgotamento do IPv4 foi o principal motor.

- **1993:** IETF reconhece o problema e cria o grupo IPng (IP Next Generation).
- **1998:** IPv6 é oficialmente padronizado (RFC 2460).
- **2011:** IANA (*Internet Assigned Numbers Authority*) esgota o *pool* global de endereços IPv4.
- **2012: World IPv6 Launch** (Google, Facebook, Yahoo) habilitam o IPv6 permanentemente.

Então em 2012 a IETF, junto com a Internet Society, Google, Facebook, Yahoo, Akamai e outras grandes empresas realizaram um evento coordenado chamado World IPv6 Launch.

A partir do **World IPv6 Launch**, as empresas envolvidas habilitaram o IPv6 sem suas redes/serviços e o **IPv6 passou do status de experimental para produção permanente de serviços reais**. Isso por exemplo obrigou/obriga que provedores de Internet comecem a dar suporte IPv6 aos seus clientes, ou pelo menos a se planejar para isso.

# Crescimento da Adoção Global (2012-2024)

A tabela a seguir apresenta o crescimento do uso do IPv6, desde 2012 até 2024:

Ano	Adoção Global Estimada	Contexto/Evento
2012	~1-2%	World IPv6 Launch
2014	~3-4%	Mais provedores começam a oferecer IPv6
2018	~15-20%	Aceleração significativa
2020	~25-30%	Pandemia acelera transformação digital
2022	~35-40%	Adoção em 40% em países desenvolvidos
2024	~40-45%	Brasil ultrapassa 50% em Fev/2024!

\*Fontes: Google [IPv6 Statistics](#), [ipv6.br](#) \*

# O que mantém o IPv4 vivo ainda hoje?

O IPv4 ainda é sustentado por adaptações técnicas ("gambiarras") que prolongaram sua vida útil:

- **CIDR (Classless Inter-Domain Routing)**
  - Permitiu alocação flexível, eliminando as classes A, B e C.
- **DHCP (Dynamic Host Configuration Protocol)**
  - Permitiu reutilização eficiente de IPs.
- **NAT (Network Address Translation)**
  - Permitiu que múltiplos dispositivos compartilhassem um IP público.

Embora essas soluções tenham sido eficazes em adiar o problema, elas introduziram complexidades e limitações que **o IPv6 foi projetado para eliminar isso** e veremos isso posteriormente.

O **NAT** por exemplo, cria pontos de passagem obrigatórios e exige a alteração constante de pacotes, o que **torna o processo de roteamento de pacotes mais complexo**.

# Objetivos do IPv6

O IPv6 não foi criado apenas para resolver o problema de escassez de endereços, mas também para oferecer melhorias significativas:

1. **Espaço de endereçamento expandido:** de  $2^{32}$ , para  $2^{128}$ ;
2. **Simplificação do cabeçalho IP:** O que torna o processamento das informações mais rápido;
3. **Autoconfiguração (SLAAC):** Possibilidade real de autoconfiguração dos dispositivos de rede, sem depender de um servidor DHCP;
4. **Segurança nativa:** IPsec é parte obrigatória - vem pronto para uso, mas não quer dizer que está ativado (precisa configurar).
5. **Melhor suporte para QoS:** permitindo classificação de tráfego pelo fluxo de rede que ele pertence, sem por exemplo verificar os outros cabeçalhos (ex. outras camadas);
6. **Eliminação de *broadcast*:** na verdade ainda há algo similar ao *broadcast*, mas é um *multicast* especial que fica confinado dentro de um enlace de rede;
7. **Mobilidade - MIPv6:** suporte nativo para dispositivos móveis. Isso permite que você continue com um IP, mesmo que você saia de uma rede e entre em outra (ex. saída de uma rede 5G para uma WiFi).

# Endereçamento IPv6: Estrutura

O IPv6 é estruturado da seguinte forma:

- **128 bits** de comprimento.
- Divididos em **8 grupos** (hextetos) de **16 bits** cada.
- Escritos em notação **hexadecimal**.
- Separados por dois pontos ( `:` ).

## Exemplo de um endereço IPv6 completo:

```
2001:0db8:85a3:0000:0000:8a2e:0370:7334
```

## Exemplo do IPv6 do DNS do Google:

```
2001:4860:4860:0000:0000:0000:0000:8888
```

## Exemplo do IPv6 do DNS do CloudFlare:

```
2606:4700:4700:0000:0000:0000:0000:1111
```

Os dois últimos exemplos são os antigos `8.8.8.8` e `1.1.1.1` do IPv4, que são IPs do Google e CloudFlare, respectivamente.

# Endereçamento IPv6: Regras de Simplificação

Temos duas regras para tornar os endereços mais legíveis:

## 1. Omissão de Zeros à Esquerda

- Zeros à esquerda em cada grupo podem ser omitidos.
- Original: 2001:0db8:00a3:0000:0000:08a2:0370:7334
- Reduzido: 2001:db8:a3:0:0:8a2:370:7334

## 2. Compressão de Sequências de Zeros (::)

- Uma sequência contínua de grupos de zeros pode ser substituída por `::`.
- **Esta regra SÓ PODE SER USADA UMA VEZ por endereço!**
- Original: fe80:0000:0000:0000:0000:0000:0000:0001
- Reduzido: fe80::1

Então os IPs do DNS do Google e do CloudFlare na forma contraída ficam da seguinte forma:  
2001:4860:4860::8888 e 2606:4700:4700::1111 , respectivamente.



# Prefixo, CIDR e Estrutura Global

Assim como no IPv4, o IPv6 utiliza a **notação CIDR**:

`2001:db8:abcd:0012::/64`

- `/64` = Comprimento do Prefixo (primeiros 64 bits identificam a rede).
- Restantes 64 bits = Interface ID (identificador do host).

**Estrutura padrão de um Endereço Global (GUA):**

48 bits	16 bits	64 bits
Global Prefix	Subnet ID	Interface ID
(Rede da Operadora)	(Sub-redes locais)	(Host)

**Recomendação padrão:** Usar prefixo `/64` para todas as LANs.

# Tipos de Endereços IPv6

O IPv6 define **três categorias principais**:

## 1. *Unicast*

- Identifica uma **única interface** de rede.
- (Global, Local, Link-Local).

## 2. *Multicast*

- Identifica um **grupo de interfaces**.
- Pacote é entregue a *todos* os membros do grupo.

## 3. *Anycast*

- Identifica um **conjunto de interfaces** (geralmente em roteadores).
- Pacote é entregue à interface **mais próxima** (conforme roteamento).

# 1. Endereços Unicast

Um pacote enviado para um endereço *unicast* é entregue apenas um único dispositivo de rede. A seguir são apresentados os endereços IPv6 Unicast:

## 1.1 Endereços Global Unicast Address (GUA)

Esses são os endereços IPs públicos, válidos na Internet e portanto são roteáveis na Internet. Ou seja, são os IPs utilizados para identificar hosts na Internet. Esses IPs são:

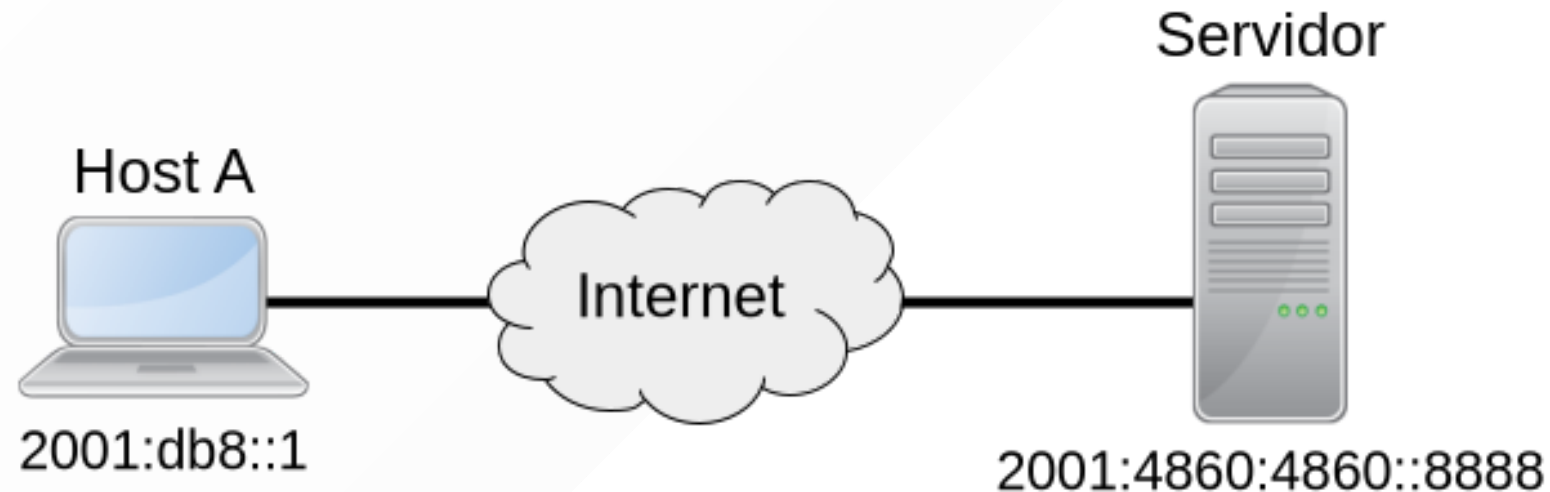
- **Prefixo:** `2000::/3` (Binário `001...`, ou seja, IPs começando com `2` ou `3`).
- **Uso:** Endereços roteáveis globalmente na Internet.
- **Equivalente IPv4:** IP Público.

Então a faixa de endereços GUA é `2000::/3`, sendo:

- `2000::`, o menor IP possível;
- `3FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF` o maior IP possível.

## 1.1 Endereços Global Unicast Address (GUA)

Então IPs que iniciam com **2** ou **3** são roteáveis na Internet. Desta forma na Internet com IPv6 só teremos essa faixa de IPs, tal como ilustra a figura a seguir:



É impreciso entender que ter um IP na faixa `2000::/3`, significa que o seu IP é acessível na Internet, o que pode ser bom (ex. fácil acesso - acesso direto), mas também pode ser ruim (segurança - no IPv4 a maioria dos IPs ficavam escondidos atrás do NAT).

## 1.2 Endereços Unique Local Address (ULA)\*\*

São endereços para identificar dispositivos em redes locais, ou seja, privadas (não Internet). Tais endereços são:

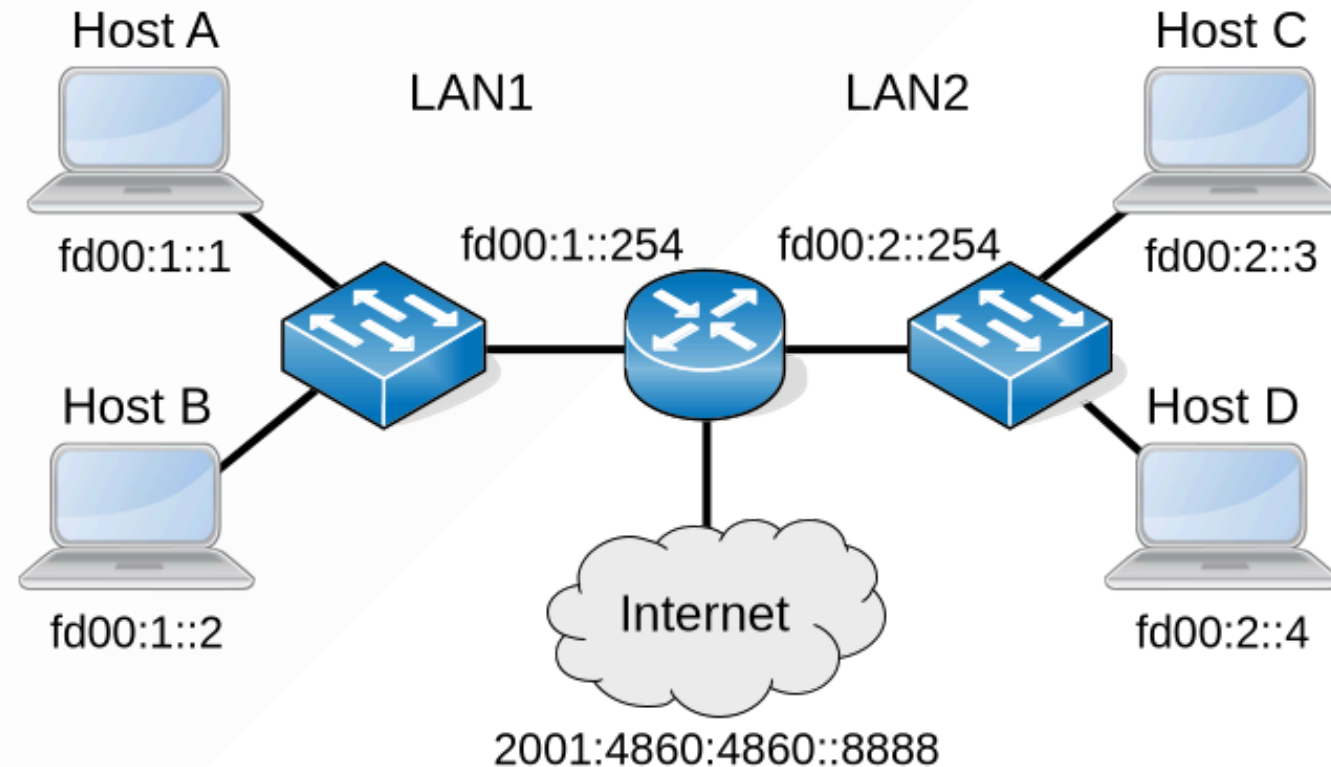
- **Prefixo:** `fc00::/7` (Na prática, usa-se `fd00::/8` ).
- **Uso:** Endereços privados não roteáveis na Internet.
- **Equivalente IPv4:** `10.0.0.0/8` , `172.16.0.0/12` , `192.168.0.0/16` .

Note que na prática o endereço recomendado para um administrador de rede utilizar na rede local é a faixa `fd00::/8` e não `fc00::/8` , pois ficou determinado o seguinte para o **8º bit** ( `1111 1100` ):

- **Se L=0**, o que vai entrar na faixa `fc00::/8` , este seria usado para endereços "**Globalmente Atribuídos**". Assim, para essa faixa uma **autoridade central poderia distribuir prefixos ULA** para garantir que essas rede fossem **100% únicas** (semelhantes aos IPs públicos), isso ajudaria na fusão de redes locais (ex. VPNs). **Todavia isso não aconteceu até agora, mas a faixa é reservada e não deve ser utilizada;**
- **Se L=1**, o que vai entrar na faixa `fd00::/8` , este seria usado para endereços "**Localmente Atribuídos**", ou seja, **é essa faixa que os administradores de rede devem utilizar para endereçar duas redes privadas.**

## 1.2 Endereços Unique Local Address (ULA)

O cenário de rede a seguir apresenta duas redes privadas, sendo essas a LAN1 e a LAN2, respectivamente com as redes: `fd00:1::/64` e `fd00:2::/64`.



É importante notar que esses IPs `fd00::/8` não são roteáveis na Internet, eles só vão ser roteáveis entre as redes privadas.

## 1.2 Endereços Unique Local Address (ULA)

É importante saber, que a rede anterior utilizou endereços bem simples e limpos para identificar a LAN1 e LAN2, que foram: `fd00:1::/64` e `fd00:2::/64`. Todavia, a recomendação não é essa, a parte que identifica essas redes poderiam ser únicas globalmente, para que em uma possível fusão/comunicação com outras redes, não houvesse conflito.

Como isso deve ser feito é estipulado pela RFC 4193, mas pode ser feito em ferramentas e *script* em sistemas como Linux, mas principalmente pode ser utilizado geradores online, tais como:

- <https://unique-local-ipv6.com>;
- <https://dnschecker.org/ipv6-address-generator.php>.

Assim, utilizando por exemplo o <https://unique-local-ipv6.com>, foi gerado o `fd0d:ba88:e700::/48`, que pode ser utilizado nas redes do exemplo da seguinte forma:

- LAN1 - `fd0d:ba88:e700:1::/64` ;
- LAN1 - `fd0d:ba88:e700:2::/64` .

Ou seja, as redes agora teriam esse início gerado automaticamente, e depois poderíamos dar a ideia de sub-rede utilizando outro "**hexteto**", no exemplo anterior: `...:1::/64` e `...:2::/64`.

## 1.3 Endereços Link-Local Address

Os endereços Link-Local (Link-Local Address) são endereços IPv6 usados exclusivamente para comunicação dentro de um único segmento de rede ou enlace local. Eles são gerados automaticamente em todas as interfaces habilitadas para IPv6 e não são roteáveis, o que significa que nunca atravessam um roteador.

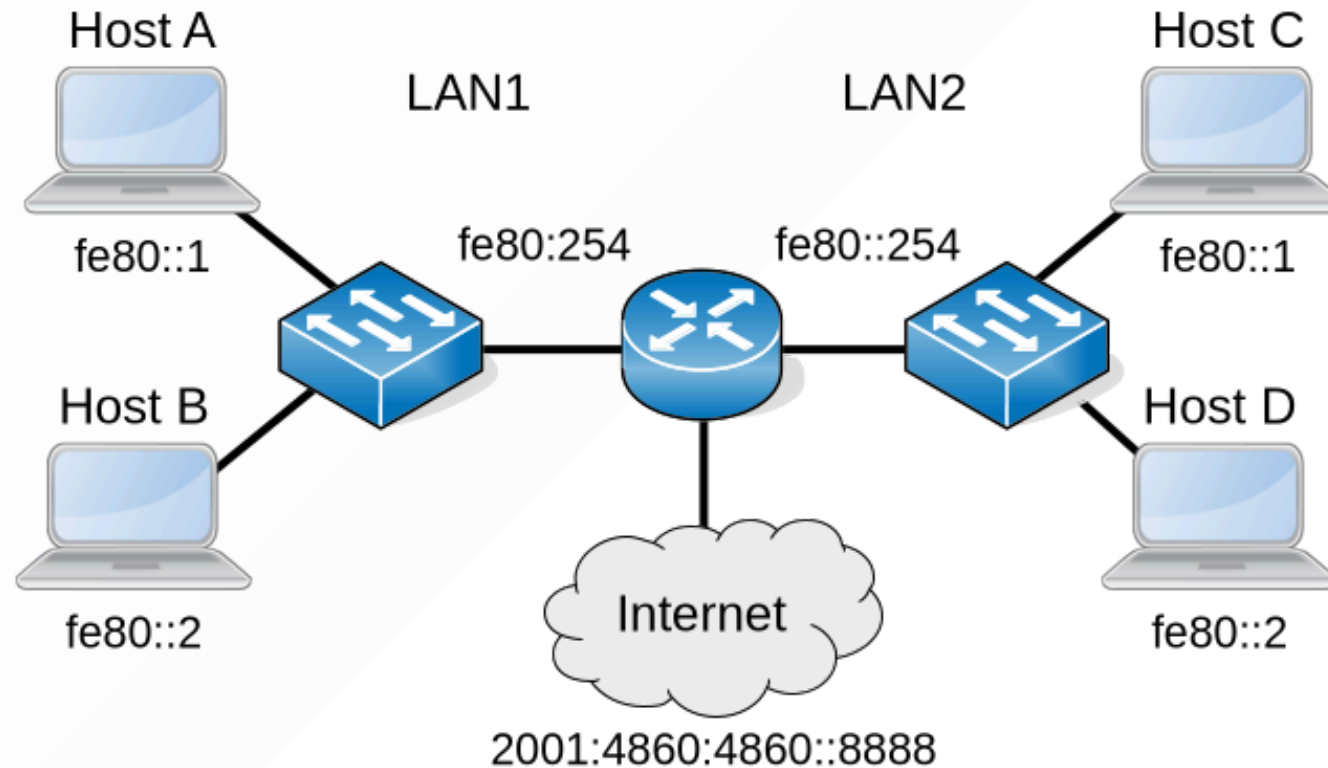
- **Prefixo:** `fe80::/10`
- **Uso:** Comunicação *apenas* dentro do mesmo link/segmento.
- **Equivalente IPv4:** APIPA `169.254.0.0/16` .
- **Importante:**
  - **Toda** interface IPv6 habilitada *automaticamente* gera um Link-Local.
  - Não atravessa roteadores.
  - Usado para autoconfiguração, descoberta de vizinhos (NDP).

Os endereços de Link-Local ( `fe80::/10` ) são fundamentais para o funcionamento de protocolos locais, como a Descobertas de Vizinhaça, auto configuração (SLAAC) e a comunicação entre dispositivos no mesmo link antes mesmo de terem endereços globais.



## 1.3 Endereços Link-Local Address

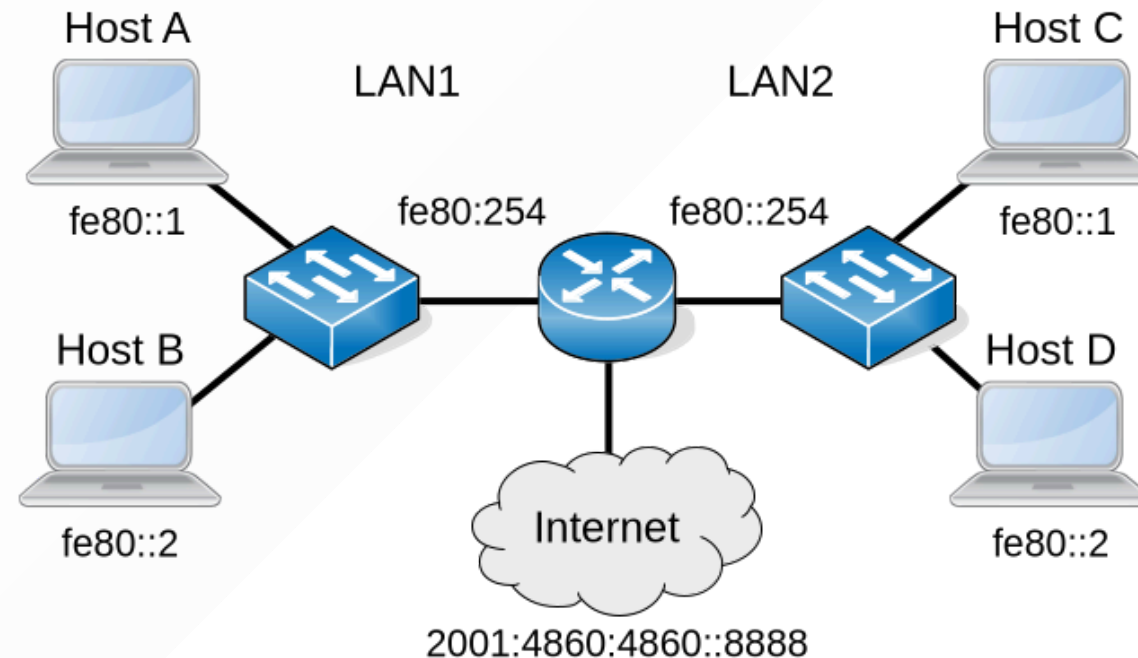
A rede a seguir apresenta a ideia do endereço Link-Local:



**Atenção:** neste exemplo utilizamos a parte que identifica o host como um hexteto fácil de identificar/ler. Todavia na prática a parte que identifica o host será gerado utilizando o endereço físico da placa de rede, somado a algum fator aleatório. Exemplo: `fe80::d019:763a:f74c:d24`

## 1.3 Endereços Link-Local Address

Outra questão importante no exemplo da rede de exemplo, é que mesmo que exista o endereço `fe80::1` na LAN1, com o Host A e `fe80::1` na LAN2, com o Host C. Esses não entraram em conflito, pois os **pacotes com endereço `fe80::/10` não são roteados de uma rede para outra.**



**Inclusive o próprio roteador pode ter o mesmo IP para redes distintas que ele esteja conectado.** É claro que isso traz confusão e estranheza para nós, mas o sistema de rede sabe separar isso.

## 1.4 Endereço de Loopback

O IPv6 tal como o IPv4 também tem um endereço de *loopback*, também conhecido como localhost. Assim o endereço `::1` é utilizado para comunicação inter-processos e pode ser também por exemplo, utilizado para testes locais - neste contexto local significa sem sair do próprio host, já que este tipo de endereço não é roteável para fora do host.

- **IPv6:** `::1/128`
- **Equivalente IPv4:** `127.0.0.1`



É importante observar que no IPv6 o único IP localhost é o `::1` e não uma faixa de IPs, tal como era no IPv4.

## 1.5 Endereço Não Especificado (*Unspecified*)

O endereço `::/128` (composto por 128 bits zerados) é chamado de Endereço Não Especificado (*Unspecified Address*). Então tal IP:

- **IPv6:** `::/128`
- **Equivalente IPv4:** `0.0.0.0`
- Usado por hosts que ainda não têm IP.

Como endereço IP de origem, o `::/128` é usado apenas para indicar a ausência de endereço, como um host enviando um pedido DHCPv6.

## 1.6 Endereço da rota padrão

O `::/128` não é a rota padrão, pois a rota padrão é o endereço `::/0` - bem na verdade a parte do IP é a mesma, todavia é preciso notar que o endereço não especificado utiliza o prefixo `\128`, enquanto a rota padrão utiliza o prefixo `\0`.

## 1.7 Endereço IPv4 Mapeado

O Endereço IPv6 Mapeado para IPv4 (*IPv4-Mapped IPv6 Address*) é um tipo especial de endereço IPv6 usado como uma técnica de transição, projetado para que aplicações possam suportar IPv4 e IPv6 simultaneamente com mais facilidade.

Esse tem o formato `::ffff:/96`, onde os últimos 32 bits do endereço IPv6 são preenchidos com o endereço IPv4 (por exemplo: `::ffff:10.1.1.1`).

- **Formato:** `::ffff:192.168.1.1`
- Usado em técnicas de transição de IPv4 para IPv6.

O uso mais comum é em sistemas que operam em Pilha Dupla (Dual Stack), especialmente em servidores. Dessa forma, o software de um servidor (a aplicação) pode ser programado para lidar apenas com endereços IPv6, e o sistema operacional cuida de "traduzir" qualquer conexão IPv4 recebida para esse formato especial, permitindo compatibilidade total em um ambiente de Pilha Dupla.

## 2. Endereços Multicast

É um endereço IP que **representa um grupo de dispositivos** (chamados de "membros"). Quando um pacote é enviado para um endereço *multicast*, a rede o entrega de forma eficiente a todos os membros que se registraram para receber essa comunicação, em vez de enviar cópias individuais para cada um. É uma comunicação do tipo "**um-para-muitos**". Os endereços *multicast* no IPv6 são:

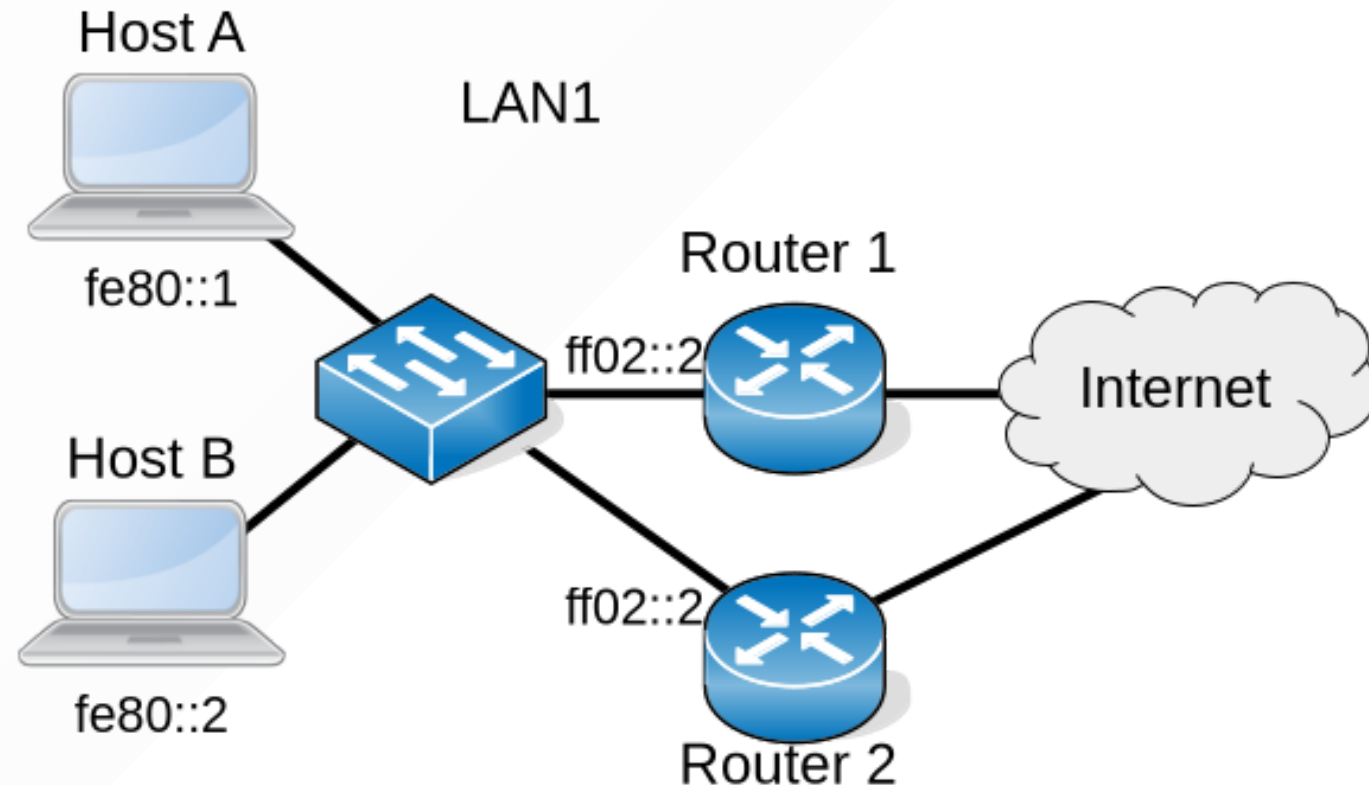
- **Prefixo:** `ff00::/8` ;
- **Uso:** Substitui o ***broadcast*** para comunicação um-para-muitos.;
- **Escopo:** Pode ser local, global, etc.

### Exemplos mais importantes (Link-Local):

- `ff02::1`
  - **All-Nodes** (Todos os Nós).
  - O verdadeiro substituto do *Broadcast*.
- `ff02::2`
  - **All-Routers** (Todos os Roteadores).
  - Usado por hosts para encontrar roteadores (NDP).

## 2.1 Endereços Multicast Link-Local

Esse tipo de endereço *multicast*, representado pelo `ff02::`, é um *multicast* que não é encaminhado para a Internet, ou seja, é utilizado apenas dentro de redes locais, para por exemplo descobrir roteadores na rede.



## 2.1 Endereços Multicast Link-Local

Neste exemplo, **os hosts da LAN 1**, podem enviar uma mensagem para o `ff02::2`, que representa um *multicast* para roteadores locais, assim essa mensagem **seria entregue tanto para o Router 1 quanto para o Router 2**.

Ainda levando em conta o exemplo anterior, no IPv6 **os dois roteadores vão responder como roteador a um pedido por exemplo do Host A**. Então **o Host A vai instalar rotas padrão (*Default Router List*) para o Router 1 quanto para o Router 2**. Desta forma, o Host A pode inicialmente escolher enviar pacotes utilizando o Router 1, mas se esse falhar ele pode enviar para o Router 2.

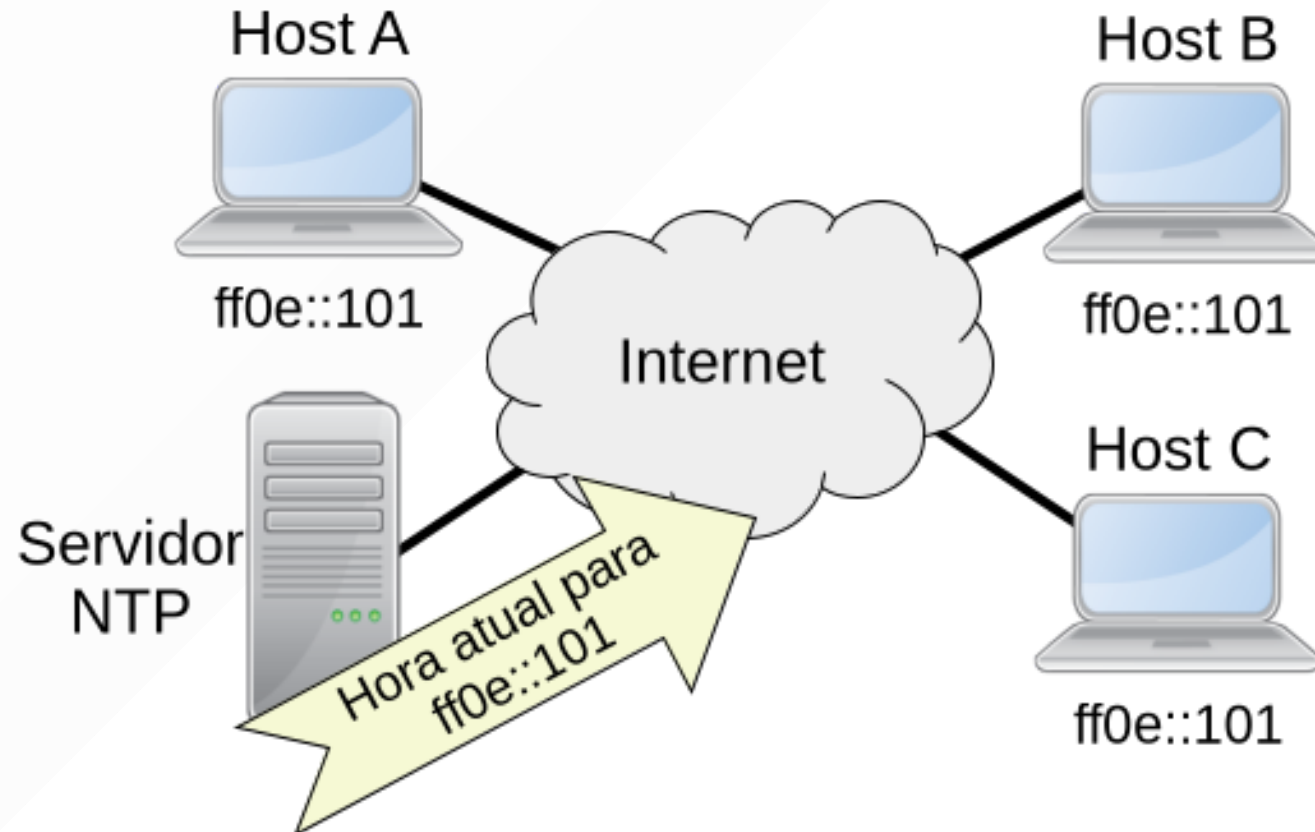
É importante observar então, que **o IPv6 implementa por padrão redundância e tolerância a falhas (*failover*) para o roteador padrão**, o que não existia por padrão no IPv4.

**O administrador da rede então pode determinar qual roteador da rede tem preferência (*Default Router Preference*)**, configurando esses com prioridades: **alta, média e baixa**. Então um roteador marcado com preferência alta, será escolhido como padrão em comparação a outros com preferência média e baixa.



## 2.2 Endereços Multicast Global

O *multicast* com endereços IPv6 globais existem, mas na prática **não são tão utilizados**. Um exemplo de uso: Um serviço de sincronização de data/hora (NTP - *Network Time Protocol*), pode atualizar os seus clientes utilizando o endereço `ff0e::101`. Desta forma, todos os clientes deste serviço que utilizam tal IPv6 receberiam atualização do horário vindas do "servidor".



## 2.3 Diferenciando melhor Multicast Global e Local

Vimos anteriormente um endereço *multicast* de **Link-Local** ( `ff02::2` ) e **Global** ( `ff0e::101` ). Assim, vamos entender e analisar melhor esses endereços:

- **Prefixo** ( `ff` ): é o prefixo universal que identifica qualquer IPv6 *multicast*.
- **Flags** ( `0` ): Campo "*flags*", no qual 0 geralmente indica u endereço permanente bem conhecido (*well-known*), alocado pelo IANA.
- **Escopo** ( `2` ou `e` ):
  - `2` , se for IPv6 *multicast* Link-Local;
  - `e` , se for IPv6 *multicast* global.
- **Group ID** ( `::*` ): O restante do endereço, é utilizado para identificar um grupo específico (**Group ID**), no exemplo que usamos:
  - `::2` : para identificar roteadores em um Link-Local;
  - `::101` : para identificar um serviço de sincronização de data/hora.

### 3. Endereços *Anycast*

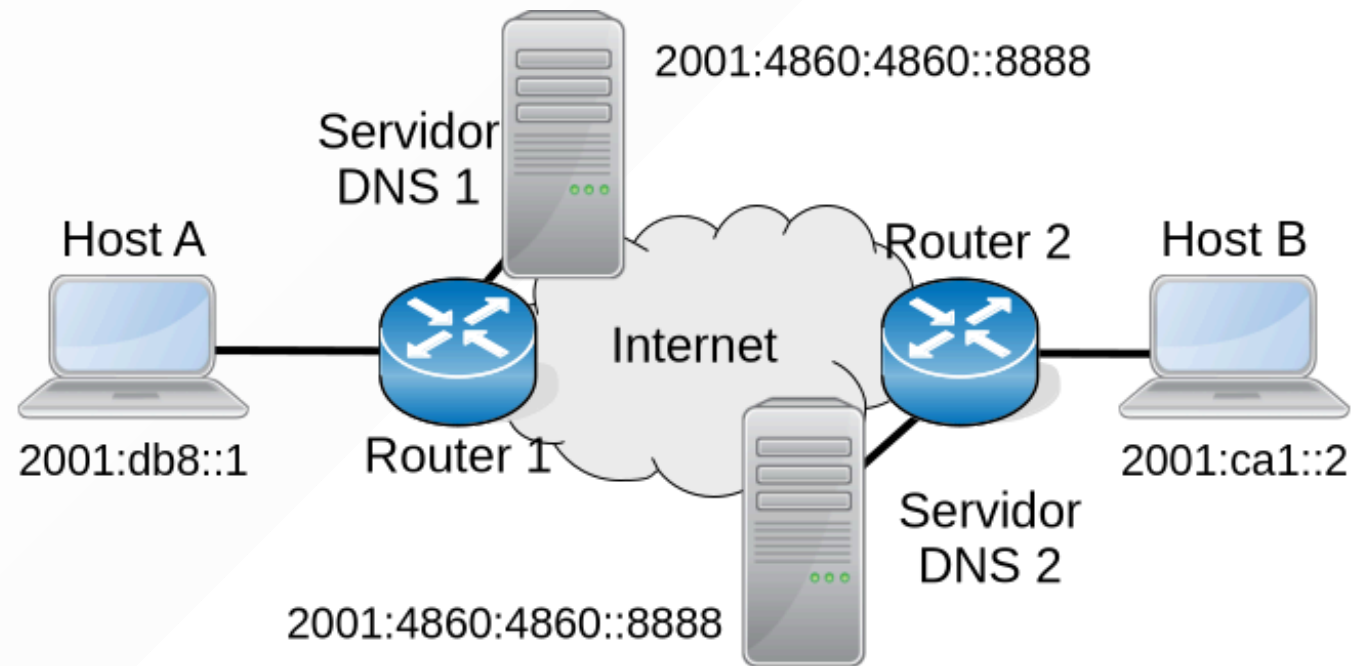
*Anycast* é um endereço IP único atribuído a múltiplos dispositivos. A rede entrega qualquer pacote enviado a esse endereço ao dispositivo "mais próximo" (com a menor métrica de roteamento), permitindo redundância e balanceamento de carga.

- **Formato:** Usa o mesmo espaço de endereçamento Unicast.
- **Conceito:** Um mesmo endereço IP (ex: `2001:db8::1`) é configurado em *vários dispositivos* (roteadores) em locais diferentes.
- **Funcionamento:** O pacote é entregue ao dispositivo "mais próximo" (menor métrica de roteamento).
- **Uso:** Alta disponibilidade e balanceamento de carga (Ex: Servidores DNS).

O *anycast* é um conceito de roteamento (e não um recurso exclusivo de um protocolo) que já era amplamente utilizado no IPv4, sendo o `8.8.8.8` o exemplo clássico.

### 3. Endereços *Anycast*

A rede a seguir apresenta o uso de *anycast* para resolução de nomes (DNS), neste caso os dois servidores de nomes têm o mesmo IP `2001:4860:4860::8888`. Desta forma, quando o Host 1 tentar resolver um nome através do IP `2001:4860:4860::8888` o pacote com este pedido será enviado para o Servidor DNS 1, que pela figura está mais perto deste host, já se o Host 2 fizer o mesmo, ele será levado ao Servidor DNS 2.



## O Fim do IP de *Broadcast*

O IPv4 usa ***Broadcast*** (ex: 192.168.1.255 para uma rede 192.168.1.0/24 ):

- **Problema:** Isso pode ser considerado "caro" e ineficiente, pois força *toda* interface de rede no enlace a parar e processar o pacote na CPU, mesmo que a informação não seja relevante.

Então, o IPv6 substitui isso pelo **Multicast**.

- Muito mais inteligente: o processamento só ocorre se o host faz parte do grupo *multicast*.
- **ff02::1 (All-Nodes):** O novo "***broadcast***" para todos. No endereço *multicast* All-Nodes, quando um host é atribuído a uma rede, ele automaticamente (através do sistemas operacional e/ou software de rede IPv6) fará parte do grupo *multicast*.
- **ff02::2 (All-Routers):** Um "***broadcast***" seletivo, apenas para roteadores, sem interromper os outros hosts. Neste *multicast*, só receberão informações destinadas ao endereço ff02::2 os roteadores, ou seja, não são todos os dispositivos que vão processar pacotes *multicast* para roteadores.

É importante lembrar que essas mensagens *multicast* não são enviados de um enlace para outro.

# O Fim do "IP de Rede"!

O IPv6 quebra a analogia do "Endereço de Rede" (primeiro IP) do IPv4 em dois conceitos:

## A. O Prefixo de Rede (Usado para a Rota)

- No IPv4, `192.168.1.0/24` é a Rota.
- No IPv6, `2001:db8:cafe:1::/64` é a **Rota**.
- Este valor (com o `::/64`) é o registro na tabela de roteamento.

## B. O Primeiro IP (O Endereço *Anycast* da Sub-rede)

- No IPv4, `192.168.1.0` é inutilizável.
- No IPv6, `2001:db8:cafe:1::` (sem o `/64`) é um IP válido, mas reservado para **Subnet-Router *Anycast***.
- É um endereço *anycast* compartilhado por *todos* os roteadores no enlace.

Para quem vem do IPv4: No IPv6 o "IP da Rede" ainda identifica a rota (ex. `2001:db8:cafe:1::/64`). A grande mudança é que, ao contrário do IPv4, onde o IP da Rede era inútil (ex. `192.168.1.0` em uma rede `/24`), no IPv6 agora ele tem uma função real: **é o endereço *Anycast* dos roteadores**.

## Resumo da Comparação (Rede e *Broadcast*)

Conceito IPv4	O que era	Análogo no IPv6	Como Funciona
<b>Endereço de Rede</b>	192.168.1.0	<b>Prefixo de Rede</b>	Identifica a rota. Ex: 2001:db8:cafe:1::/64
(Primeiro IP)	(Reservado)	<b>Subnet-Router Anycast</b>	Primeiro IP ( :: ). Usado para falar com o roteador mais próximo.
<b>Endereço de Broadcast</b>	192.168.1.255	<b>Multicast All-Nodes</b>	ff02::1 . Entrega para todos os nós no enlace.
(Novo Conceito)	(Não existia)	<b>Multicast All-Routers</b>	ff02::2 . Entrega apenas para os roteadores no enlace.

Então no IPv6 temos que ter em mente que:

- O primeiro IP, que identifica a rota, agora também tem a função de Subnet-Router *Anycast*.
- O IP de *Broadcast* não existe mais, o que "economiza" o último IP de cada rede. Ele foi substituído por *Multicast* Link-Local ( ff02::1 e ff02::2 ), que nunca são roteados para fora da rede.

# Cálculo da Quantidade de Hosts em redes IPv6

Calcular o número de endereços disponíveis é mais direto no IPv6.

## Em IPv4:

- Fórmula:  $2^n - 2$  (onde  $n$  = bits de host).
- Subtraímos **2** endereços:
  1. Endereço de Rede (primeiro IP).
  2. Endereço de *Broadcast* (último IP).

## Em IPv6 (para uma sub-rede /64):

- Bits de host ( $n$ ):  $128 - 64 = 64$  bits.
- Total de endereços IPs:  $2^{64}$  (~18 quintilhões).
- Fórmula de Hosts:  $2^{64} - 1$
- Subtraímos apenas **1** endereço:
  1. O "**Subnet-Router Anycast**" ( `...::` ).
- Não subtraímos o broadcast. O último IP ( `...:ffff:ffff:ffff:ffff` ) é um host válido!



# Tabela Comparativa de Tipos de Endereços IPv4/IPv6

Função	IPv4	IPv6	Observações
Loopback	127.0.0.1/8	::1/128	Teste local da pilha TCP/IP
Não especificado	0.0.0.0	::	Ausência de endereço
Link-local	169.254.0.0/16 (APIPA)	fe80::/10	Comunicação local (obrigatório no IPv6)
Endereços privados	10.0.0.0/8 , 172... , 192...	fd00::/8 (ULA)	Não roteáveis na Internet
Endereços públicos	Endereços roteáveis	2000::/3 (GUA)	Roteáveis globalmente
Broadcast	255.255.255.255	Não existe	Substituído por <i>multicast</i>
Rota padrão	0.0.0.0/0	::/0	<i>Default gateway</i>
Documentação	192.0.2.0/24	2001:db8::/32	Exemplos e testes

# Métodos de Configuração IPv6

O IPv6 oferece múltiplas formas de configuração de endereços:

## 1. Configuração Estática/Manual

- Administrador configura IP, prefixo e *gateway*.
- Ideal para servidores, roteadores e infraestrutura.

## 2. SLAAC (*Stateless Address Autoconfiguration*)

- Autoconfiguração **sem estado** (sem servidor).
- Host gera seu próprio IP usando informações do Roteador.

## 3. DHCPv6 *Stateful*

- Similar ao DHCP do IPv4.
- Servidor fornece o IP completo e todas as opções (DNS, etc.).

## 4. DHCPv6 *Stateless*

- Híbrido: **SLAAC** fornece o IP + **DHCPv6** fornece o resto (DNS, etc.).

## Mensagens de Autoconfiguração de Rede: RS e RA

Antes de continuar e entender os métodos de configuração IPv6 é bom saber que em grande parte ele depende do ***Neighbor Discovery Protocol (NDP)***, que é um tipo de mensagem do **ICMPv6**.

Assim, para conseguir a configuração de rede de forma automática, o ICMPv6 envia em sua área de dados **dois tipos de mensagens** sendo essas:

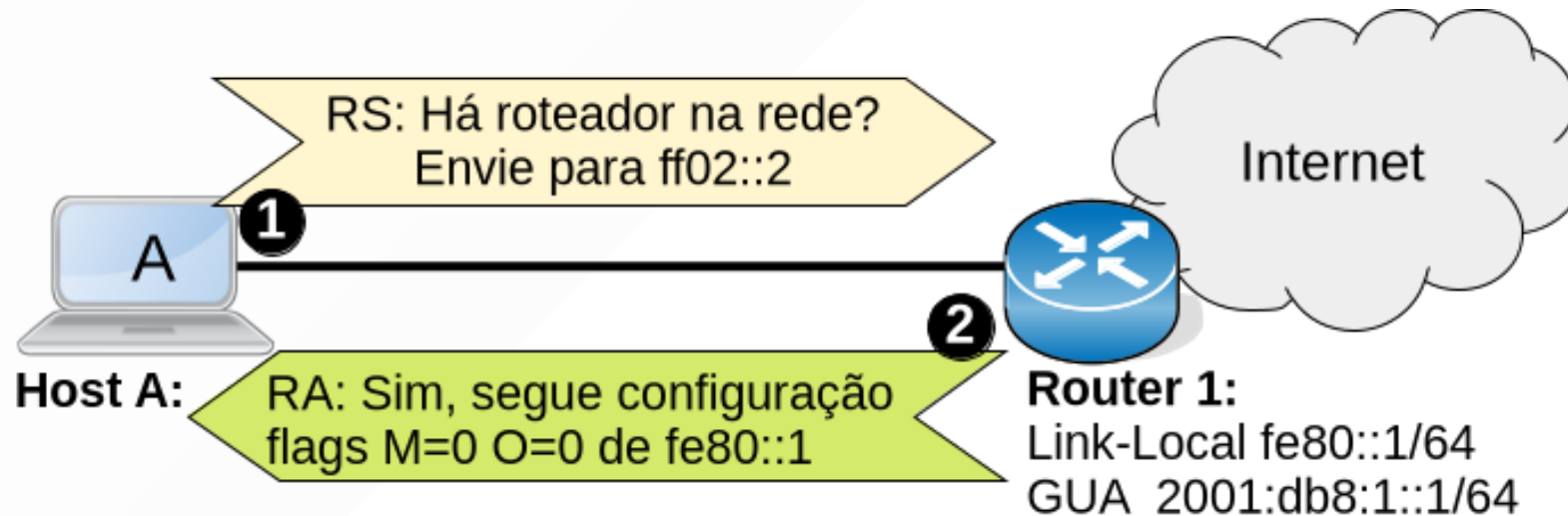
- ***Router Solicitation (RS)***, ou Solicitação de Roteador, é uma mensagem ICMPv6 (Tipo 133) que os hosts enviam **para descobrir roteadores na rede local**. Ela é enviada para o endereço *multicast* `ff02::2` (*All-Routers*) quando o host é ativado ou precisa de configuração.
- ***Router Advertisement (RA)***, ou Anúncio de Roteador, é uma mensagem ICMPv6 (Tipo 134) que um roteador envia para anunciar sua presença e fornecer os parâmetros de configuração da rede. Ela é enviada periodicamente para o *multicast* `ff02::1` (*All-Nodes*) ou como uma resposta direta a um RS. O RA contém informações cruciais como os prefixos de rede (ex: `.../64`), o MTU e as **flags M e O** (que definem o método de configuração).

Não é intenção desse material abordar em detalhes o ICMPv6.

## Mensagens de Autoconfiguração de Rede: RS e RA

As mensagens ICMPv6 RS e RA são então encapsuladas dentro dos *datagramas* IPv6 e posteriormente são enviadas à Camada de Enlace

A figura a seguir apresenta a ideia do envio das mensagens de autoconfiguração de rede com o envio inicial de RS, por parte do Host A e na sequência o Router 1 respondendo com uma mensagem ICMPv6 do tipo RA.



Dentro da mensagem RA segue as **flags M (Managed) e O (Other)**, que indicam qual tipo de autoconfiguração de rede será utilizada para o Host solicitante.

## Um pouco mais sobre as Flags M e O do ICMPv6 RA

Então as *flags* M (*Managed*) e O (*Other*) são bits de informação (0 ou 1) que o roteador envia dentro das mensagens de **Router Advertisement (RA)** para dizer aos hosts na rede como eles devem obter seus endereços IP e outras configurações.

Sendo as possíveis combinações dessas:

- **M=0 e O=0** - Neste caso seria o **SLAAC Puro**:
  - M=0, significa **não gerenciado**;
  - O=0, significa **nenhuma outra informação** de rede será repassada.
- **M=1** - Neste cenário a configuração será **via DHCPv6 Stateful** e a *flag* O não importa, ou seja, os hosts da rede não vão usar SLAAC, mas sim um servidor DHCP que dará todas as configurações de rede;
- **M=0 e O=1** - Neste cenário a configuração será via SLAAC, mas usando um **modo híbrido**, ou seja, ele **vai utilizar o prefixo de rede para auto-atribuir seu IP, mas pegará outras informações**, tais como IP do servidor DNS utilizando, por exemplo **via um servidor DHCPv6 Stateless**.

Agora vamos continuar e ver em detalhes os métodos de configuração de rede do IPv6.

## 2. SLAAC (*Stateless Address Autoconfiguration*)

É o método de autoconfiguração padrão do IPv6.

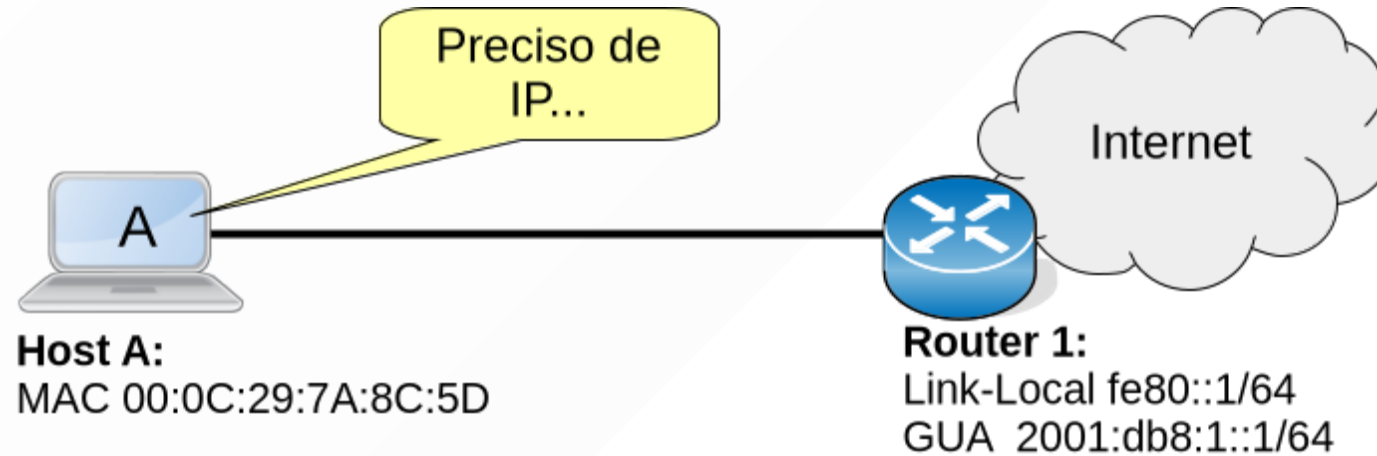
### Funcionamento:

1. Host liga e envia um **Router Solicitation (RS)** para `ff02::2` (All-Routers).
2. Roteador responde com um **Router Advertisement (RA)** contendo:
  - Prefixo da rede (ex: `2001:db8:1::/64` ).
  - Flags de configuração (M - Managed, O - Other), que o roteador envia dentro das mensagens RA, em resposta aos RS para informar aos hosts da rede como eles devem obter as configurações de rede.
3. Host combina o **Prefixo** (64 bits) com um **Interface ID** (64 bits) para formar seu endereço.

Endereço = [ Prefixo do Roteador ] + [ ID da Interface ]

## 2. SLAAC (*Stateless Address Autoconfiguration*)

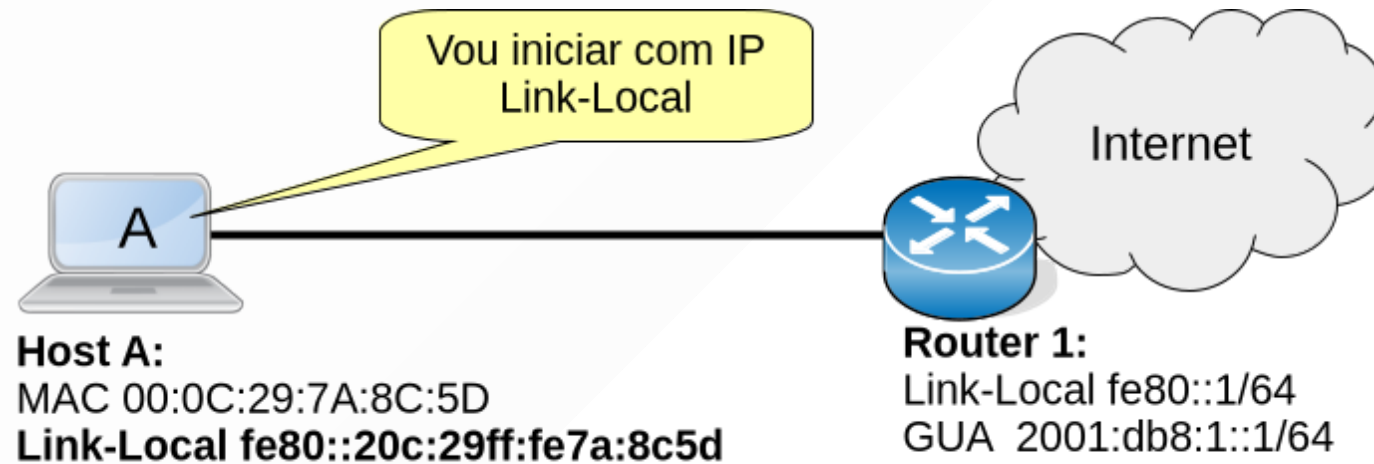
Vamos entender como é feita a configuração do SLAAC:



**Primeiro** o computador inicia sem configuração de rede e então inicia o processo de obtenção ou auto configuração de rede.

## 2. SLAAC (Stateless Address Autoconfiguration)

O segundo passo é se auto atribuir um IP Link-Local na rede `fe80::/64` e alocar a parte do host com seu endereço.MAC (ID da interface de rede).

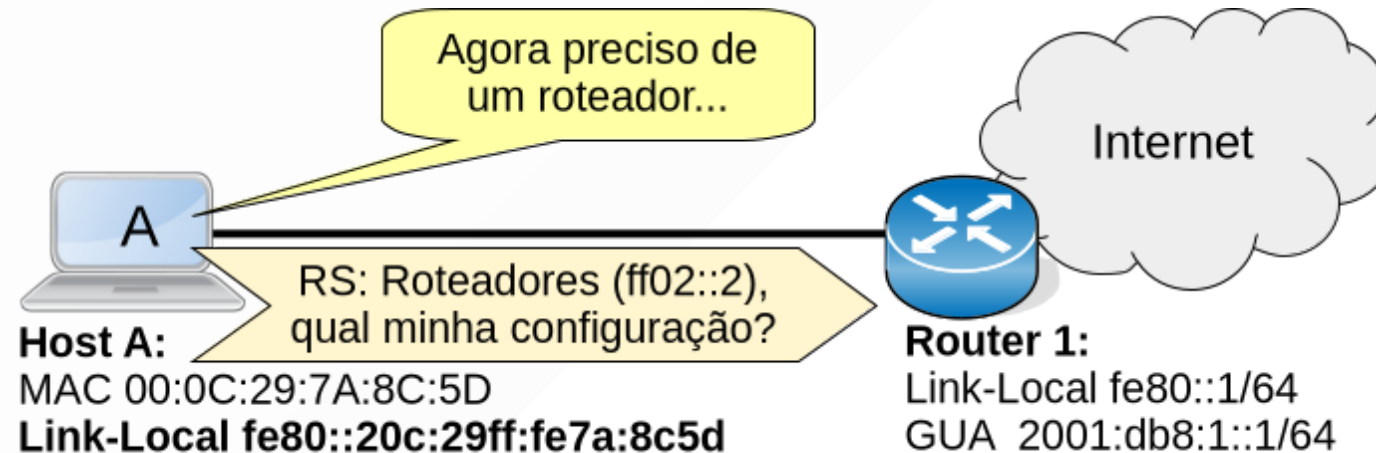


Com esse IP de Link-local agora o host pode acessar a rede local.



## 2. SLAAC (Stateless Address Autoconfiguration)

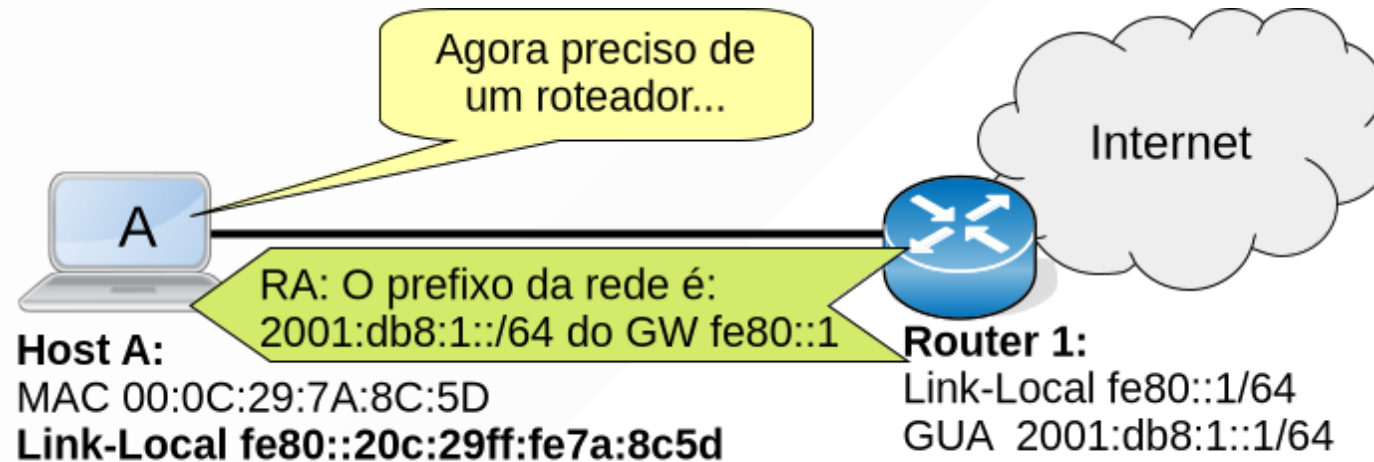
Com um IP Link-Local único na rede, ele inicia o processo de descobrir se há roteadores na rede, para isso ele envia uma mensagem RS (Router Solicitation), para o endereço *multicast* dos roteadores, ou seja, para `ff02::2`.



Se houver roteador esse vai responder com um RA (Router Advertisement).

## 2. SLAAC (Stateless Address Autoconfiguration)

No cenário de exemplo, há um roteador e esse responde com o prefixo de rede `2001:db8:1::/64` e também informando o seu endereço Link-Local unicast, que no caso do exemplo é o `fe80::1/64`.



Então o host pega o prefixo de rede enviado pelo roteador, adiciona a parte do host o seu endereço MAC e instala a rota padrão como sendo para o *gateway* `fe80::1`.

Assim, essas imagens ilustram brevemente os passos de auto-configuração usando o método SLAAC do IPv6. É importante notar que há outras variações para esses passos, mas a ideia básica é essa.

## Como o SLAAC gera o Interface ID (Host ID)?

O host precisa gerar seus próprios 64 bits. Existem **3 formas para isso**, sendo essas:

### 1. EUI-64 (Extended Unique Identifier)

- Usa o Endereço MAC (48 bits) para gerar o ID (64 bits).
- Também é inserido o **FFFE** ao MAC e inverte o 7º bit.
- **Desvantagem:** Expõe o MAC do fabricante o que gera preocupações de privacidade.

Por exemplo:

- Supondo que o endereço MAC de um dado host é **00:0C:29:7A:8C:5D** ;
- Então o EUI-64 para esse host seria gerado como: **020c:29ff:fe7a:8c5d** ;

Assim, levando em consideração o exemplo anterior, olha-se para o primeiro grupo, que é **00** em hexa e **00000000** em binário, como o penúltimo valor (7º bit) é zero, invertendo ele vai ficar com o valor **1**, ou seja: **00000010** em binário ou **02** em hexadecimal, o que resulta no valor **020c:29ff:fe7a:8c5d**.

# Como o SLAAC gera o Interface ID (Host ID)?

## 2. Privacy Extensions (RFC 4941)

- Gera um Interface ID **aleatório e temporário**.
- Muda periodicamente (ex: a cada 7 dias).
- **Vantagem:** Protege a privacidade (dificulta rastreamento).
- **Desvantagem:** Dificulta o gerenciamento (*logs*, regras de firewall, *troubleshooting*), pois o IP do host está sempre mudando.
- **Padrão em sistemas:** Windows, macOS, iOS e Android.

## 3. Aleatório Estável (Linux)

- Gera um ID aleatório, o IP é aleatório (esconde o MAC), mas não muda.
- **Vantagem:** É mais fácil de gerenciar (o IP do host é estável para *logs* e ACLs), sendo um "compromisso" entre EUI-64 e privacidade total.
- **Desvantagem:** Menos privacidade que as Privacy Extensions, pois o dispositivo ainda é rastreável dentro daquela rede, pois a identificação não muda (o IP sempre será o mesmo).

### 3. DHCPv6 Stateful (Com Estado)

Funciona de forma similar ao DHCP do IPv4: controle centralizado.

#### Funcionamento:

1. Roteador anuncia no RA a flag **M=1** ("Managed").
2. Flag **M=1** instrui o host: "Não use SLAAC. Peça seu IP a um servidor DHCPv6."
3. Host envia **DHCPv6 Solicit** para **ff02::1:2** (*All-DHCP-Agents*).
4. Servidor responde com IP, *gateway*, DNS, etc.

**Vantagens:** Controle total, auditoria, gerenciamento centralizado.

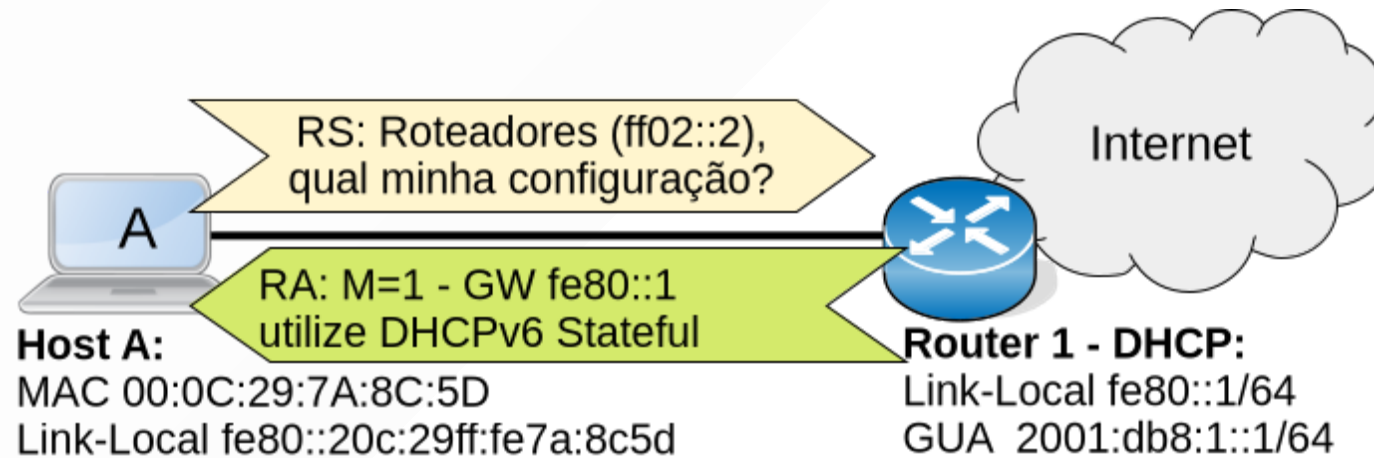
**Desvantagens:** Requer infraestrutura de servidor; Android (antigo) não suportava.

Então, essa forma de configuração, seria muito similar ao que já é utilizado no DHCP do IPv4 (o que se utiliza atualmente). Entretanto, mesmo que o Host que vai pedir sua configuração de rede tenha em sua configuração "estática" (no momento do *boot*), que ele deve utilizar DHCP, ele ainda vai enviar o RS e se o RA informar que a configuração deve ser outra (SLAAC, por exemplo) o host deve acatar a configuração enviada pelo roteador via RA.

### 3. DHCPv6 Stateful (Com Estado)

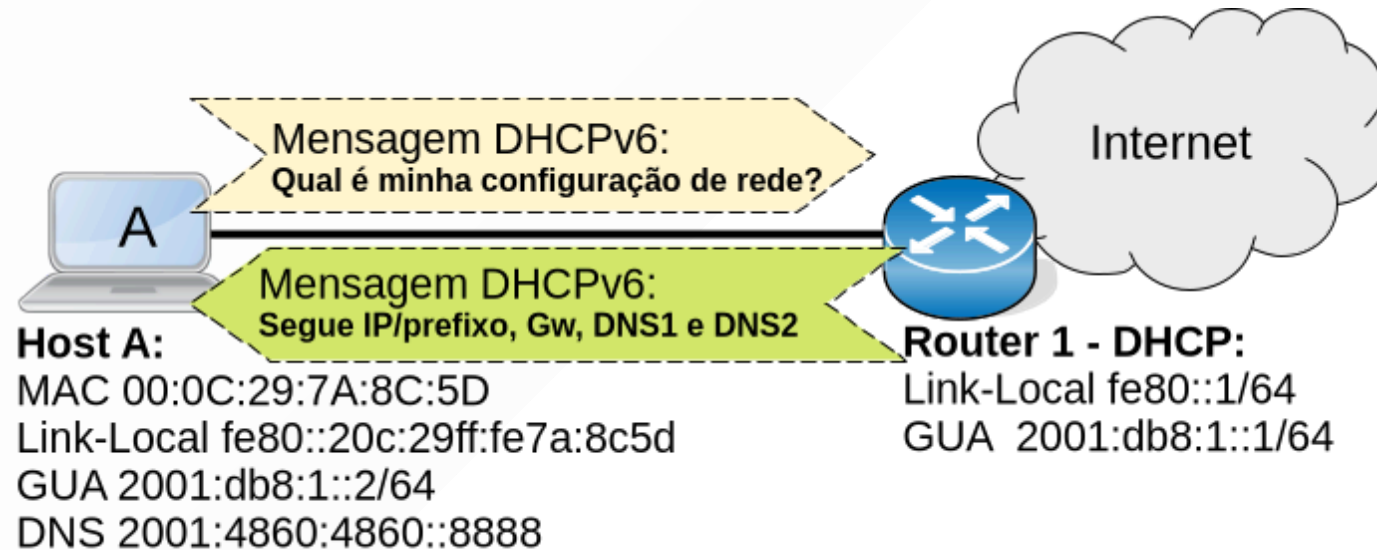
A figura a seguir apresenta um exemplo de uso do DHCPv6 Stateful. Neste cenário o Router 1 é o roteador da rede e também o servidor DHCPv6.

Assim, tudo inicia com o host enviando um RS, então o roteador deve lhe informar que ele vai utilizar DHCP Stateful, isso é feito pelo RA com a *flag* M=1.



### 3. DHCPv6 Stateful (Com Estado)

Depois do RA confirma que o tipo de configuração de rede deve ser o DHCPv6 Stateful. O host envia uma mensagem DHCPv6 solicitando sua configuração de rede, e se tudo correr bem, o servidor DHCP vai responder.



É necessário saber, que mesmo que o IPv6 consiga fazer o endereçamento ser um servidor DHCPv6 *Stateful*, o uso de um servidor para controlar IPs é uma prática muito recomendável, principalmente em redes maiores e mais complexas, para por, exemplo fornecer IPs conhecidos para servidores, roteadores, etc. Ou seja, sem o DHCPv6 Stateful a rede tende a ficar menos organizada.

## 4. DHCPv6 Stateless (Sem Estado)

Uma combinação inteligente: SLAAC para o IP, DHCPv6 para o resto.

### Funcionamento:

1. Roteador anuncia no RA as flags **M=0** e **O=1** ("Other").
2. **M=0** diz: "Use SLAAC para gerar seu próprio IP."
3. **O=1** diz: "Mas, depois disso, procure um servidor DHCPv6 para pegar o *resto* das informações (DNS, domínio, etc.)."

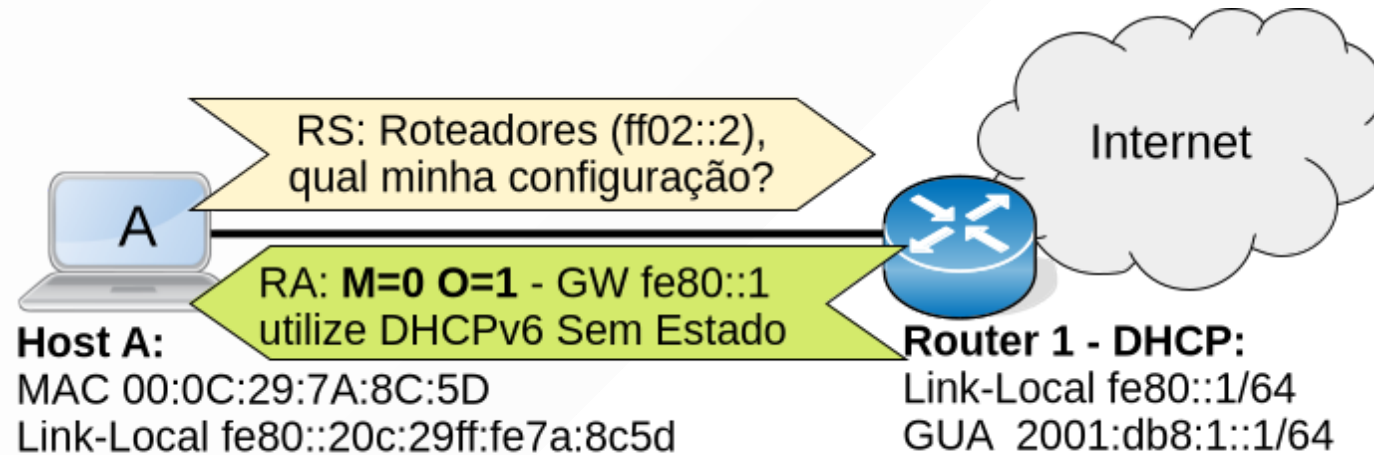
### Vantagens:

- Simplicidade do SLAAC para endereçamento.
- Flexibilidade do DHCPv6 para opções (como DNS).



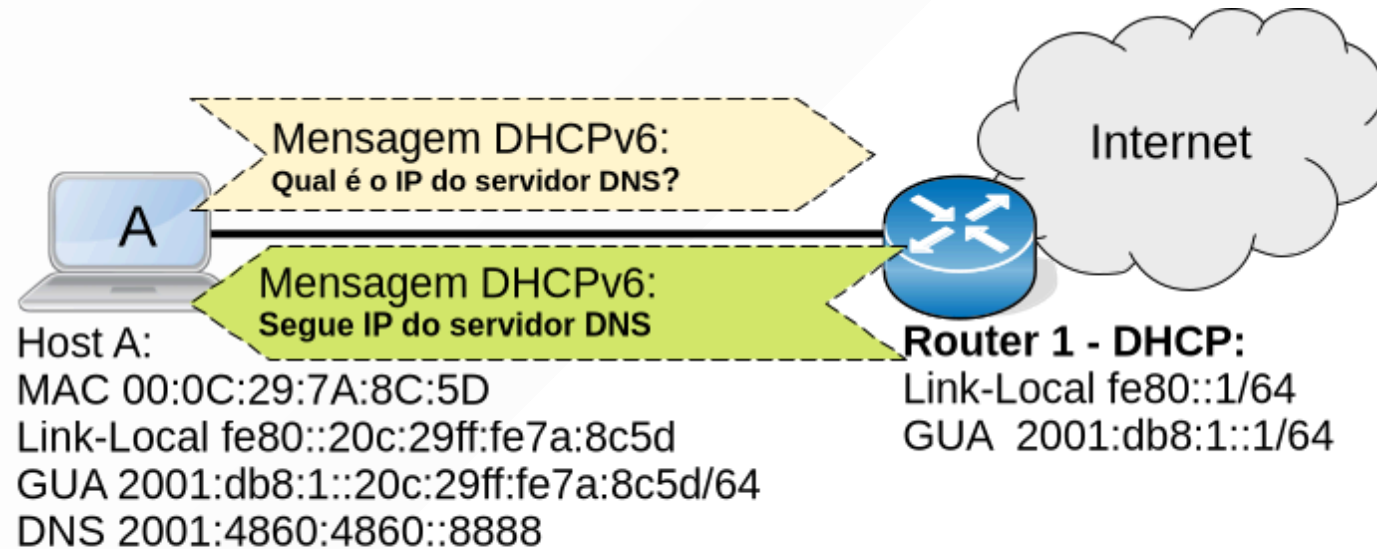
## 4. DHCPv6 Stateless (Sem Estado)

Esse método inicia tal como foi explicado no DHCPv5 Stateful, ou seja, com o envio do RS e a espera pela resposta de um RA, só que agora para esse método o **RA deve enviar as *flags* M=0 e O=1**.



## 4. DHCPv6 Stateless (Sem Estado)

Então o host obtém do roteador que o IP pode ser *stateless*, mas é preciso obter outras configurações de rede via DHCP, tal como obter o IP do servidor DNS para conseguir acessar a rede por nomes. Veja o exemplo da figura a seguir:



Na figura anterior, o IP dado é obtido de forma *stateless* e portanto é adicionado o endereço MAC ao final do prefixo da rede obtido via RA. Todavia é enviada uma mensagem DHCPv6 para obter o IP do servidor DNS. Essa resposta é dada pelo servidor DHCP, tal informação é instalada no host.

## Comparação dos Métodos de Configuração

Método	Controle	Complexidade	Privacidade	Escalabilidade	Uso Típico
<b>Estático</b>	Total	Baixa	Alta	Baixa	Servidores, Roteadores
<b>SLAAC + EUI-64</b>	Baixo	Baixa	Baixa	Alta	Laboratórios, Servidores
<b>SLAAC + Privacy</b>	Baixo	Baixa	<b>Alta</b>	Alta	Clientes (Windows, Mac, Mobile)
<b>DHCPv6 Stateful</b>	<b>Total</b>	Alta	Média	Alta	Corporativo (Controle Total)
<b>DHCPv6 Stateless</b>	Médio	Média	Média	Alta	Híbrido (Melhor dos dois)

## Conclusão

Então, este material mostrou como funciona o IPv6, principalmente o endereçamento IPv6 e as formas de configuração de rede de computadores via IPv6. Tudo isso demonstra que o IPv6 já está pronto para trazer melhorias para a Internet, entretanto é necessário refletir sobre:

- **Desafios:** Custo de atualização, complexidade do *Dual-Stack*, inércia ("se o IPv4 funciona, por que mudar?").
- **Por que migrar?** O esgotamento do IPv4 é real. CGNAT é problemático. IoT e 5G *exigem* IPv6.
- **Então uma estratégia para a mudança para o IPv6 seja:**
  1. Planejamento (treinamento).
  2. Piloto (redes de teste).
  3. Implementação **Dual-Stack**.
  4. Migração para IPv6-Only (futuro).

Uma coisa parece certa: **o IPv6 vem ganhando mais força para substituir o IPv4 e ser o protocolo da Internet.**

Mas parece que isso nunca acontece em 100%. :-p

## Referências e Leitura Adicional

- **RFC 8200:** Internet Protocol, Version 6 (IPv6) Specification
- **RFC 4291:** IP Version 6 Addressing Architecture
- **RFC 4861:** Neighbor Discovery for IP version 6 (NDP)
- **RFC 4862:** IPv6 Stateless Address Autoconfig (SLAAC)
- **RFC 4193:** Unique Local IPv6 Unicast Addresses (ULA)
- **Google IPv6 Statistics:** <https://www.google.com/intl/en/ipv6/statistics.html>
- **IPv6.br:** <https://ipv6.br/>

# IPv6

## Introdução



**Prof. Luiz Arthur Feitosa dos Santos**

[luiz.arthur.feitosa.santos@gmail.com](mailto:luiz.arthur.feitosa.santos@gmail.com)/[luizsantos@utfpr.edu.br](mailto:luizsantos@utfpr.edu.br)

**Fim... Obrigado!**

# Licença de Uso

Este material está licenciado sob a **Creative Commons Atribuição 4.0 Internacional (CC BY 4.0)**.



## Você tem o direito de:

- **Compartilhar** — copiar e redistribuir o material em qualquer suporte ou formato.
- **Adaptar** — remixar, transformar e criar a partir do material para qualquer fim, mesmo que comercial.

## Sob os seguintes termos:

- **Atribuição** — Você deve dar o crédito apropriado, prover um link para a licença e indicar se foram feitas alterações.

[https://creativecommons.org/licenses/by/4.0/deed.pt\\_BR](https://creativecommons.org/licenses/by/4.0/deed.pt_BR)