

# Sockets IPv6 para C/C++ - Criando e Portando Aplicações

Rodrigo Regis dos Santos  
[rsantos@nic.br](mailto:rsantos@nic.br)

25 de Junho de 2009

*fisl*10  
10º Fórum Internacional  
Software Livre  
A tecnologia que liberta

**IPv6.br**  
A Nova Geração do  
Protocolo Internet

# Agenda

- O CGI.br e o NIC.br.
- O Protocolo IPv6.
- Interoperabilidade
- Portando Aplicações.
  - Funções de conexão;
  - Estruturas de dados;
  - Funções para tradução de nomes e endereços;
  - Funções de conversão de endereços;
  - Logs, Interface e novas funcionalidades do IPv6.
- Onde Aprender Mais?

# Sobre o CGI.br

## Comitê Gestor da Internet no Brasil.

- Criado em maio de 1995 pela Portaria Interministerial N° 147 de 31/05/1995, alterada pelo Decreto Presidencial N° 4.829 de 03/09/2003
- Responsável pela coordenação e integração dos serviços Internet no país
- Modelo *multistakeholder* composto por membros do governo, e membros eleitos dos setores empresarial, terceiro setor e da comunidade acadêmica.
- Não é órgão do governo
- Não tem personalidade jurídica

# Principais atribuições do CGI.br

- **Fomentar** o desenvolvimento de serviços Internet no Brasil
- **Recomendar** padrões e procedimentos técnicos operacionais para a Internet no Brasil
- **Coordenar** a atribuição de endereços Internet (IPs) e o registro de nomes de domínios usando .br
- **Coletar, organizar e disseminar** informações sobre os serviços Internet – indicadores e estatísticas

# Sobre o NIC.br

## **Núcleo de Informação e Coordenação do Ponto BR**

- Entidade civil, sem fins lucrativos, criada em 2003 e começando a atuar em 2005 (delegação do CGI.br)
- Conselho de Administração composto por 7 membros: 3 do governo, escolhidos entre os componentes do CGI.br; 4 do setor privado indicados pelo CGI.br.
- Assembléia Geral formada pelo pleno do CGI.br
- Braço executivo do Comitê Gestor da Internet no Brasil
- Coordena as atividades do Registro, do CERT, do CETIC e do CEPTRO.
- Abriga o escritório W3C Brasil.

# O Protocolo IPv6

## IPv4

- Definido pela RFC791 (1981)
- Inicialmente para uso acadêmico
- 1990 - passa a ser explorado comercialmente
- Crescimento da Internet
- Esgotamento dos Endereços
- Aumento da Tabela de Roteamento
- Segurança
- Qualidade de Serviço (QoS)

## IPv6

- Definido pela RFC2460 (1998)
- Solução para todos esses problemas?
- Novo formato de cabeçalho
- Maior espaço para endereçamento
- Mecanismos de Auto-configuração
- Segurança incorporada
- Melhor suporte a QoS
- Tratamento da fragmentação de pacotes

# O Protocolo IPv6

- Estrutura do Cabeçalho

Cabeçalho IPv4



- Campos Inalterados
- Campos Removidos
- Nome e Posição Alterada
- Novo Campo do IPv6

Cabeçalho IPv6



**Cabeçalhos de Extensão:** *Hop-By-Hop, Destination Options, Routing, Fragmentation, Authentication, Encapsulating Security Payload.*

# O Protocolo IPv6

- Endereçamento

Um endereço **IPv4** é formado por **32 bits**.

$2^{32} = 4.294.967.296$  endereços

Um endereço **IPv6** é formado por **128 bits**.

$2^{128} = 340.282.366.920.938.463.463.374.607.431.768.211.456$  endereços

~ 56 octilhões ( $5,6 \times 10^{28}$ ) de endereços IP por ser humano.

~ 79 octilhões ( $7,9 \times 10^{28}$ ) de endereços a mais do que no IPv4.

# O Protocolo IPv6

- Endereçamento

O IPv6 é representado por 8 campos de endereços de 16 bits (em forma hexadecimal), separados por dois pontos

**2001:0db8:0000:130F:0000:0000:087C:140b**

**2001:db8:0:130F::087C:140b**

Prefixos

- Como o CIDR (IPv4)
- Exemplo: **2001:db8:12::/48**

URL

- [http://\[2001:DB8:CAFE::20\]:8080](http://[2001:DB8:CAFE::20]:8080)
- [http://\[2001:DB8:CAFE::20\]/index.htm](http://[2001:DB8:CAFE::20]/index.htm)

# O Protocolo IPv6

- Endereçamento
- Categorias:
  - **Unicast** – Isso equivale a um endereço IP de destino.
    - *Global Unicast; Link-Local (FE80::/64); Unique-Local (FC00::/7); Unspecified; Loopback; IPv4-mapped (::FFFF:wxyz).*
  - **Multicast** – O *Multicast* será enviado para vários destinos (pense em um broadcast para uma sub-rede). Todas as máquinas, de acordo com suas configurações, vão processar a informação (**FF00::/8**).
  - **Anycast** – O *Anycast* será enviado para vários destinos, mas, no fim, será entregue ao primeiro dispositivo que estiver no caminho de roteamento.

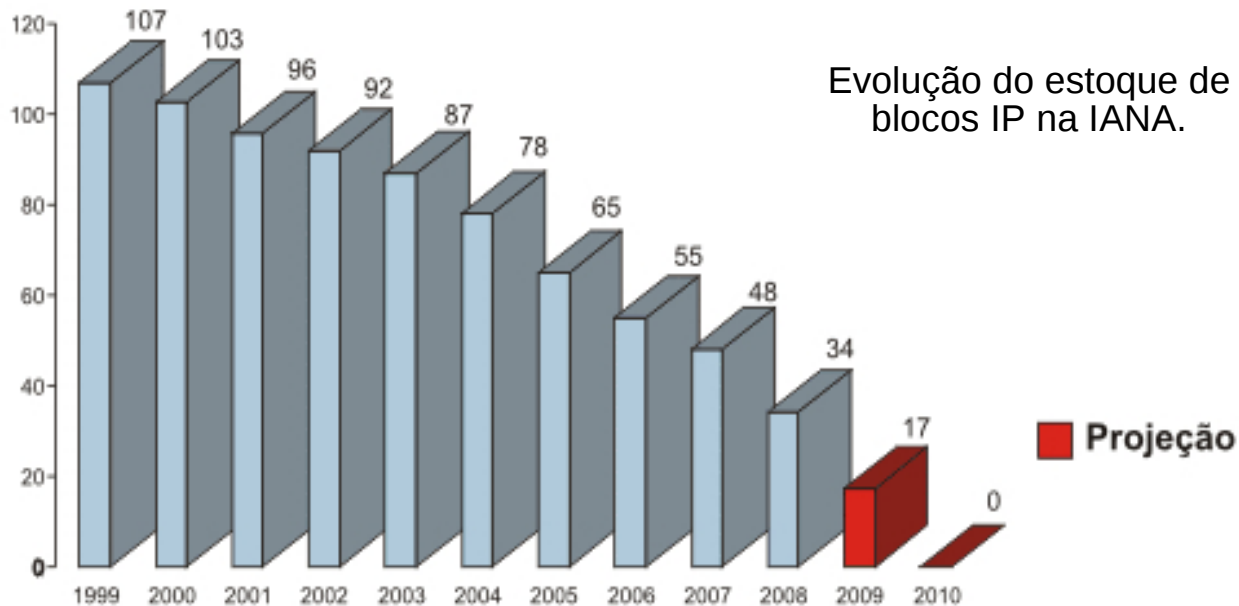
Não existe mais **Broadcast**.

# O Protocolo IPv6

- FUNCIONALIDADES E SERVIÇOS
  - Descoberta de Vizinhaça (*Neighbor Discovery*)
  - Fragmentação
  - *Jumbograms*
  - Suporte a Mobilidade
  - QoS
  - SEGURANÇA
    - IPSec
    - SEND

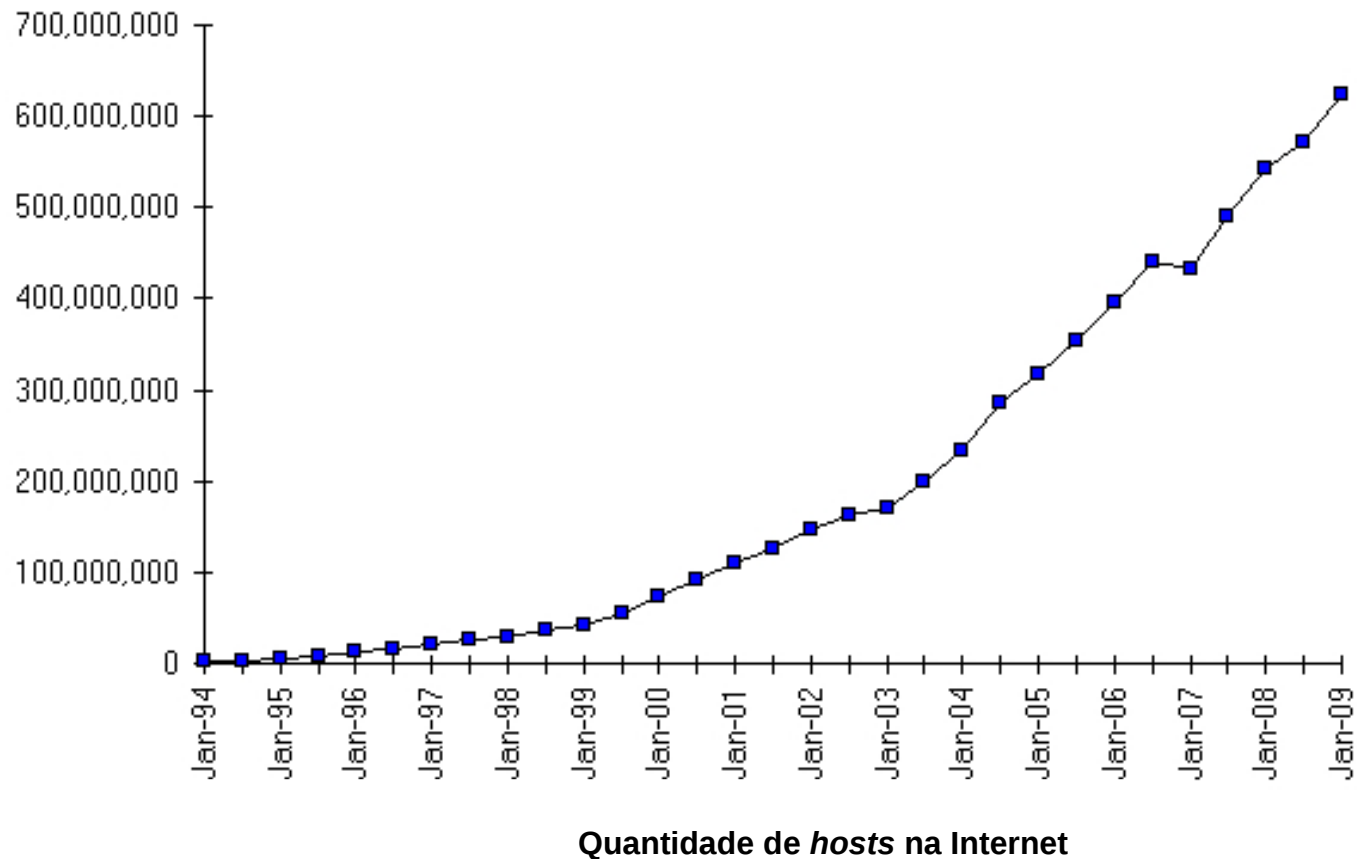
# Por que utilizar IPv6 hoje?

- Hoje existem apenas 30 blocos /8 livres na IANA, ou seja, apenas 11% do total; Previsões atuais apontam para um esgotamento desses blocos em 2010; O estoque dos RIRs deve durar 2 ou 3 anos a mais.



# Por que utilizar IPv6 hoje?

- A Internet continua crescendo



# Soluções

## Soluções paliativas:

- 1992 - IETF cria o grupo ROAD (*ROuting and ADdressing*).
  - CIDR (RFC 4632)
    - ♦ Fim do uso de classes = blocos de tamanho apropriado.
    - ♦ Endereço de rede = prefixo/comprimento.
    - ♦ Agregação das rotas = reduz o tamanho da tabela de rotas.
- DHCP
  - Alocações dinâmicas de endereços.
- NAT + RFC 1918
  - Permite conectar toda uma rede de computadores usando apenas um endereço válido na Internet, porém com várias restrições.

# Soluções

## ▪ NAT

### • Vantagens:

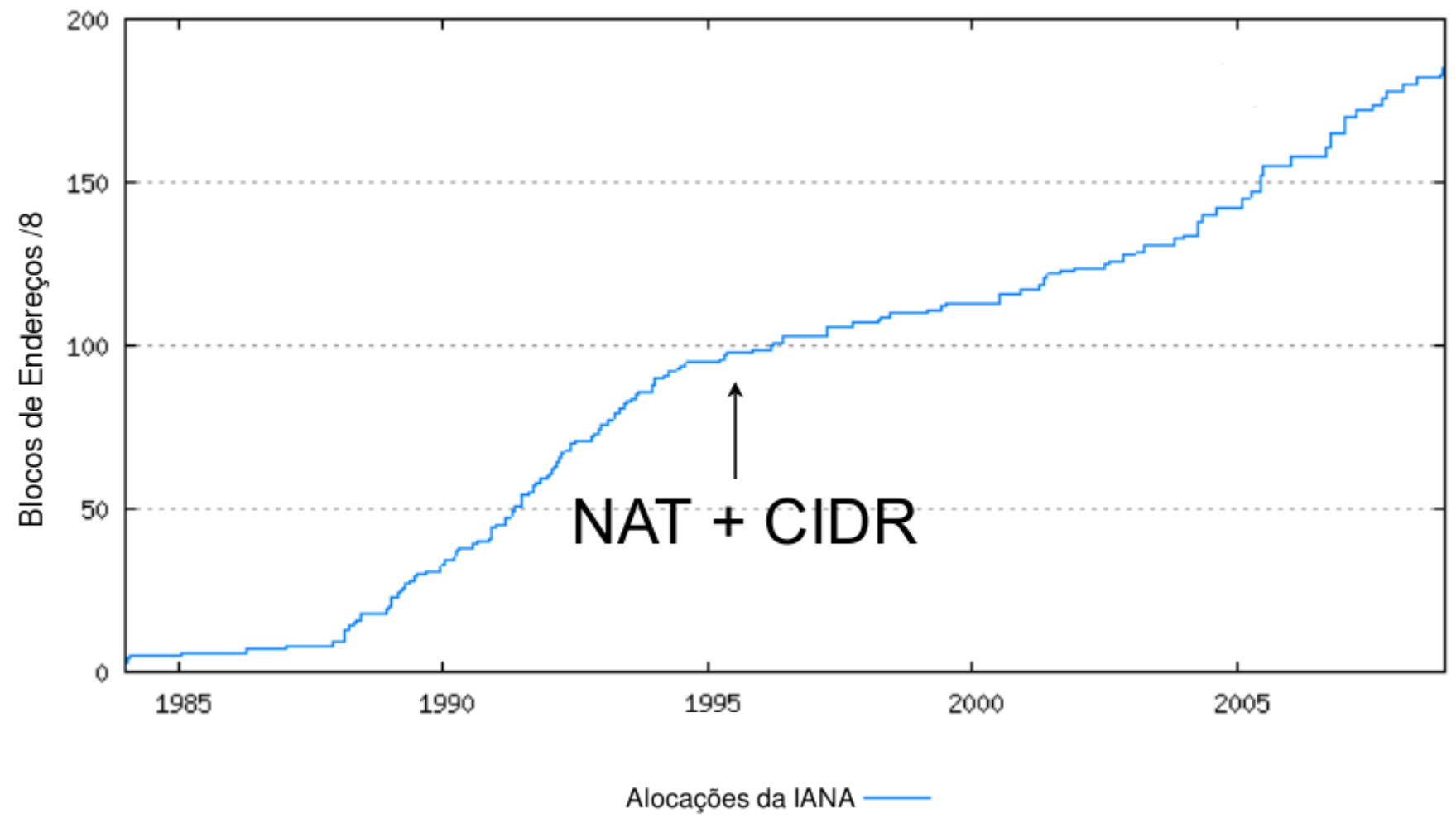
- ♦ Reduz a necessidade de endereços públicos;
- ♦ Facilita a numeração interna das redes;
- ♦ Oculta a topologia das redes;
- ♦ Só permite a entrada de pacotes gerado em resposta a um pedido da rede.

### • Desvantagens:

- ♦ Quebra o modelo fim-a-fim da Internet;
- ♦ Dificulta o funcionamento de uma série de aplicações;
- ♦ Não é escalável;
- ♦ Aumento do processamento no dispositivo tradutor;
- ♦ Falsa sensação de segurança;
- ♦ Impossibilidade de se rastrear o caminho do pacote;
- ♦ Impossibilita a utilização de algumas técnicas de segurança como IPSec.

# Soluções

**Soluções paliativas: Queda de apenas 14%**



# Quais os riscos da não implantação do IPv6?

- Embora ainda seja pequena, a utilização do IPv6 tem aumentado gradativamente;
- Porém precisa avançar ainda mais;
- A não implementação do IPv6 irá:
  - Impedir o surgimento de novas redes;
  - Diminuir o processo de inclusão digital o reduzindo o número de novos usuários;
  - Dificultar o surgimento de novas aplicações;
  - Aumentar a utilização de técnicas como a NAT.
- O custo de não implementar o IPv6 poderá ser maior que o custo de implementá-lo;
- Provedores Internet precisam inovar e oferecer novos serviços a seus clientes.

# Interoperabilidade

- Migração ou Implantação?
- O IPv4 vai continuar a funcionar por vários anos.
- Implantação a curto prazo...
- Transição a longo prazo:
  - IPv4...

# Interoperabilidade

- Migração ou Implantação?
- O IPv4 vai continuar a funcionar por vários anos.
- Implantação a curto prazo...
- Transição a longo prazo:
  - IPv4...
  - Ilhas IPv6 conectadas por túneis IPv4...

# Interoperabilidade

- Migração ou Implantação?
- O IPv4 vai continuar a funcionar por vários anos.
- Implantação a curto prazo...
- Transição a longo prazo:
  - IPv4...
  - Ilhas IPv6 conectadas por túneis IPv4...
  - Ilhas IPv4 conectadas por túneis sobre uma Internet... predominantemente IPv6...

# Interoperabilidade

- Migração ou Implantação?
- O IPv4 vai continuar a funcionar por vários anos.
- Implantação a curto prazo...
- Transição a longo prazo:
  - IPv4...
  - Ilhas IPv6 conectadas por túneis IPv4...
  - Ilhas IPv4 conectadas por túneis sobre uma Internet... predominantemente IPv6...
  - **IPv6!**

# Interoperabilidade

- Nesta fase de transição, o que é importante ter em mente na hora de desenvolver uma nova aplicação?
  - IPv6 não é compatível com o IPv4.
  - Alguns nós nunca migrarão para IPv6.
  - Não haverá uma “data da virada”.
  - A transição ocorrerá de forma gradual.
  - Esta mudança deve ser transparente para os usuários finais.

# Interoperabilidade

- Ferramentas de transição
  - Pilha Dupla: provê o suporte a ambos os protocolos no mesmo dispositivo.
  - Tunelamento: permite o trafego de pacotes IPv6 sobre estruturas de rede IPv4.
    - Tunnel Broker / 6to4 / ISATAP / Teredo
  - Tradução: permite a comunicação entre nós com suporte apenas a IPv6 com nós que suportam apenas IPv4.
    - SIIT / BIS / BIA / TRT / SOCKS64 / ALG

# Portando Aplicações

- A grande maioria das aplicações ainda suportam apenas IPv4.
- Algumas mudanças poderão ser realizadas automaticamente apenas utilizando *scripts*.
- Porém muitas necessitarão de uma revisão detalhada dos códigos fontes.
- Com o que é preciso se preocupar?
  - Funções de conexão;
  - Estruturas de dados;
  - Funções para tradução de nomes e endereços;
  - Funções de conversão de endereços.

# Portando Aplicações

- Funções de conexão
  - Seqüência típica de tarefas em um servidor IPv4
    - `socket` – abre um socket
    - `bind` – cria a ligação entre o endereço IP e o socket
    - `listen` – começa a ouvir uma determinada porta
    - `accept` – espera por conexões
    - `read` e/ou `write` se TCP
    - `recvfrom` e/ou `sendto` se UDP
  - Seqüência típica de tarefas em um cliente IPv4
    - `socket` – abre um socket
    - `connect` – conecta com um servidor
    - `read` e/ou `write` se TCP
    - `recvfrom` e/ou `sendto` se UDP

# Portando Aplicações

- Funções de conexão
  - Seqüência típica de tarefas em um servidor IPv6
    - `socket` – abre um socket
    - `bind` – cria a ligação entre o endereço IP e o socket
    - `listen` – começa a ouvir uma determinada porta
    - `accept` – espera por conexões
    - `read` e/ou `write` se TCP
    - `recvfrom` e/ou `sendto` se UDP
  - Seqüência típica de tarefas em um cliente IPv6
    - `socket` – abre um socket
    - `connect` – conecta com um servidor
    - `read` e/ou `write` se TCP
    - `recvfrom` e/ou `sendto` se UDP

# Portando Aplicações

- Nova Família de Endereços e Família de Protocolos
  - AF\_INET6
  - PF\_INET6
    - #defined PF\_INET6 AF\_INET6
  - Primeiro argumento da função `socket()`
  - Indica que um socket IPv6 está sendo criado.

```
s = socket(AF_INET, SOCK_DGRAM, 0); IPv4  
s = socket(AF_INET6, SOCK_DGRAM, 0); IPv6
```

- Não houve mudança na sintaxe das outras funções: `bind()`, `connect()`, `sendmsg()`, `sendto()`, `accept()`, `recvfrom()`, `recvmsg()`, `getpeername()`, `getsockname()`.

# Portando Aplicações

- Nova Estrutura de Endereços
  - Armazena um único endereço IPv6 de 128 bits.

```
struct in6_addr {  
    uint8_t s6_addr[16]; /*Endereço IPv6*/  
};
```

```
struct sockaddr_in6  
{  
    uint8_t sin6_len; /*tamanho da estrutura*/  
    sa_family_t sin6_family; /*AF_INET6*/  
    in_port_t sin6_port; /*numero da porta TCP ou UDP*/  
    uint32_t sin6_flowinfo; /*fluxo informações*/  
    struct in6_addr sin6_addr; /*endereço IPv6 de 128 bits*/  
    uint32_t sin6_scope_id; /*conjunto de interfaces de um  
escopo*/  
};
```

# Portando Aplicações

- Estrutura de endereços genérica.
  - Capaz de armazenar qualquer tipo de endereço.
  - Permite passar como argumento para algumas funções um ponteiro para esta estrutura genérica em vez de um ponteiro para um tipo específico de endereço.

```
struct sockaddr_storage {
    sa_family_t  ss_family;
    char         __ss_pad1[_SS_PAD1SIZE];
    int64_t      __ss_align;
    char         __ss_pad2[_SS_PAD2SIZE];
};
```

# Portando Aplicações

- Estrutura de endereços genérica.

```
struct sockaddr_storage ss;  
int ss_len;  
  
get_sock_addr((struct sockaddr *)&ss);  
  
switch (ss.ss_family) {  
    case AF_INET:  
        sin = (struct sockaddr_in *)&ss;  
        ss_len = sizeof(struct sockaddr_in);  
        break;  
    case AF_INET6:  
        sin6 = (struct sockaddr_in6 *)&ss;  
        ss_len = sizeof(struct sockaddr_in6);  
        break;  
    [...]  
}  
  
ret = bind(s, (struct sockaddr *)&ss, ss_len);
```

# Portando Aplicações

- Compatibilidade com nós IPv4.
  - Um socket IPv6 pode se conectar e aceitar conexões IPv4.
  - O nó IPv6 deve possuir Pilha Dupla.
  - Utiliza-se o endereço IPv4-mapeado **::FFFF:<endereço-IPv4>**.
  - Deve-se utilizar o parâmetro `AF_INET6` em aplicações que deverão suportar tanto IPv4 quanto IPv6.
  - Pode-se utilizar a marca `IN6_IS_ADDR_V4_MAPPED`.

# Portando Aplicações

- Resolução de Nomes e Endereços

```
int getaddrinfo( const char *hostname, const  
char *service, const struct addrinfo *hints,  
struct addrinfo **result );
```

- Esta função retorna um ponteiro para uma lista ligada de estruturas `addrinfo` com informações que serão utilizadas na criação dos sockets.



# Portando Aplicações

- Resolução de Nomes e Endereços

```
int getnameinfo(const struct sockaddr  
*sockaddr, socklen_t addrlen, char *host,  
socklen_t hostlen, char *service, socklen_t  
servicelen, int flags);
```

- Recebe um endereço de soquete e retorna o nome do host e o tipo de serviço.
- Pode ser utilizada para converter um endereço IPv6 da forma binária para texto e vice-versa, utilizando *flags* específicas.

# Portando Aplicações

- Macros para verificação de endereços

```
int IN6_IS_ADDR_UNSPECIFIED (const struct in6_addr *);
int IN6_IS_ADDR_LOOPBACK   (const struct in6_addr *);
int IN6_IS_ADDR_MULTICAST  (const struct in6_addr *);
int IN6_IS_ADDR_LINKLOCAL  (const struct in6_addr *);
int IN6_IS_ADDR_SITELOCAL  (const struct in6_addr *);
int IN6_IS_ADDR_V4MAPPED   (const struct in6_addr *);
int IN6_IS_ADDR_V4COMPAT   (const struct in6_addr *);
int IN6_IS_ADDR_MC_NODELOCAL(const struct in6_addr *);
int IN6_IS_ADDR_MC_LINKLOCAL(const struct in6_addr *);
int IN6_IS_ADDR_MC_SITELOCAL(const struct in6_addr *);
int IN6_IS_ADDR_MC_ORGLOCAL (const struct in6_addr *);
int IN6_IS_ADDR_MC_GLOBAL   (const struct in6_addr *);
```

# Portando Aplicações

- Mudanças na API

IPv4	IPv6
hostent	addrinfo
sockaddr_in	sockaddr_in6
sockaddr_in	sockaddr_storage
INADDR_ANY	in6addr_any
INADDR_LOOPBACK	in6addr_loopback
gethostbyname	getaddrinfo
gethostbyaddr	getnameinfo
inet_addr	getnameinfo
inet_ntoa	getnameinfo

# Portando Aplicações

- Preciso me preocupar com mais alguma coisa?
  - No lado do servidor
    - Ajustar as funções de log para que possam lidar com os IPs maiores.
    - Aumentar todas as variáveis e locais de armazenamentos de dados que lidam com IPs, inclusive nos bancos de dados.
  - No lado do cliente
    - Ajustar as funções relativas aos sockets, funções relativas aos logs e as variáveis.
    - Ajustar as funções de interface para que possam lidar com IPs maiores.

# http://www.ipv6.br

Google

br IPv6 - IPv6.br x br IPv6 - Tradução x br IPv6 - Túneis 6to4 x

← → ↻ ☆ http://ipv6.br/ ▶ 📄 🔧

Falha no carregament... SIMET SIMET IPv6 IPv6-security Segurança Medições wiki >> Outros favoritos

Comitê Gestor da Internet no Brasil Seu IP: 2001:12ff:0:10:a845:b8a:d8c5:88cb

NIC.br | CETIC.br | Antispam.br | **CEPTRO.br** > PTT.br | NTP.br | IPv6.br **Imprensa**

## IPv6.br

**Bem-vindo(a) ao IPv6.br!**

Navegue por assunto, no menu à esquerda, ou escolha a opção que melhor lhe representa a seguir:

**Usuário Final**

**Gestor**

**Governo**

**Engenheiro**

**Provedor Internet**

**O IPv6 é a nova geração do Protocolo Internet.**

Ele já vem sendo utilizado há algum tempo. Mas, agora, sua implantação deve ser acelerada. Ela é imprescindível para a continuidade do crescimento e da

**nic.br**  
Núcleo de Informação e Coordenação

**cgi.br** | Registro CERT.br

**IPv6 Brasil** Este sítio web funciona com IPv6. Se o globo estiver girando, você também já usa IPv6!

**Últimas Notícias...**

19 Nov 2008 - 13:32: [ARIN e CAIDA publicam os resultados do estudo sobre adoção do IPv6.](#)

19 Nov 2008 - 13:28: [Chamda para apresentações de trabalhos no FLIP6 e no LAC IPv6 TF](#)

13 Nov 2008 - 19:24: [Restam 36 blocos /8 de IPv4](#)

27 Oct 2008 - 19:15:

Buseca