# 1、简介

Socket是一种IPC方法，允许位于统一主机或者使用网络连接起来的不同主机上的应用程序之间交换数据。

## 通信domain

Socket存在于一个通信domain中，它确定：

1、识别出一个socket的方法（即socket地址的格式）；

2、通信范围（即是在位于同一主机上的应用程序之间还是位于同一网络连接起来的主机应用程序之间）。

现在操作系统至少支持一下domain：

UNIX(AF\_UNIX)：domain允许在同一主机上的应用程序之间进行通信。

IPv4(AF\_INET)：domain允许在使用IOv4网络连接起来的主机上应用程序之间通信。

IPv6(AF\_INET6)：domain允许在使用IPv6网络连接起来主机上应用程序之间通信。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Domain | 执行通信 | 应用间通信 | 地址格式 | 地址结构 |
| AF\_UNIX | 内核中 | 同一主机 | 路径名 | sockaddr\_un |
| AF\_INET | 通过IPv4 | IPv4网络主机 | 32位IPv4地址+16位端口号 | sockaddr\_in |
| AF\_INET6 | 通过IPv6 | IPv6网络主机 | 128位IPv4地址+16位端口号 | sockaddr\_un6 |

说明：AF=Address Family，即地址族

## Socket类型

每个socket实现至少提供两种socket：流和数据报。

流socket（SOCK\_STREAM）：提供了一个可靠的双向的字节流通信信道。含义如下：

1）可靠的：表示可以保证发送者传输的数据会完整地送达接收应用程序或者收到一个传输失败的的通知；

2）双向：表示数据可以在两个socket之间的任意方向上传输；

3）字节流：表示与管道一样不存在消息边界的概念。

数据报socket（SOCK\_DGRAM）允许数据以被称为数据报的消息的形式进行交换。在数据报socket中，消息边界得到了保留，但是数据传输是不可靠的。消息的达到可能是无序的、重复的或者根本无法到达。

在Internet domain中，数据报socket使用了用户数据报协议（UDP），而流socket则（通常）使用了传输控制协议（TCP），分别使用术语“UDP socket”和“TCP socket”。

# 2、系统调用

关键的socket系统调用包括以下几种：

socket()系统调用：创建一个新的socket；

bind()系统调用：将一个socket绑定到一个地址上，通常，服务器需要使用这个调用来将其socket绑定到一个众所周知的地址上使得客户端能够定位到该socket上；

listen()系统调用：允许一个流socket接受来自其他socket的接入连接；

accept()系统调用：在一个监听流socket上接受来自一个对等应用程序的连接，并可选地返回对等socket的地址；

connect()系统调用：建立一个与另一个socket之间的连接。

Socket I/O可以使用传统的read()和write()系统调用或使用一组socket特有的系统调用（如send()、recv()、sendto()、recvfrom()）来完成。在默认情况下，这些系统调用在I/O操作无法被立即完成时会被阻塞。通过使用fcntl() F\_SETFL操作来启用O\_NONBLOCK打开文件状态标记可以执行非阻塞I/O。