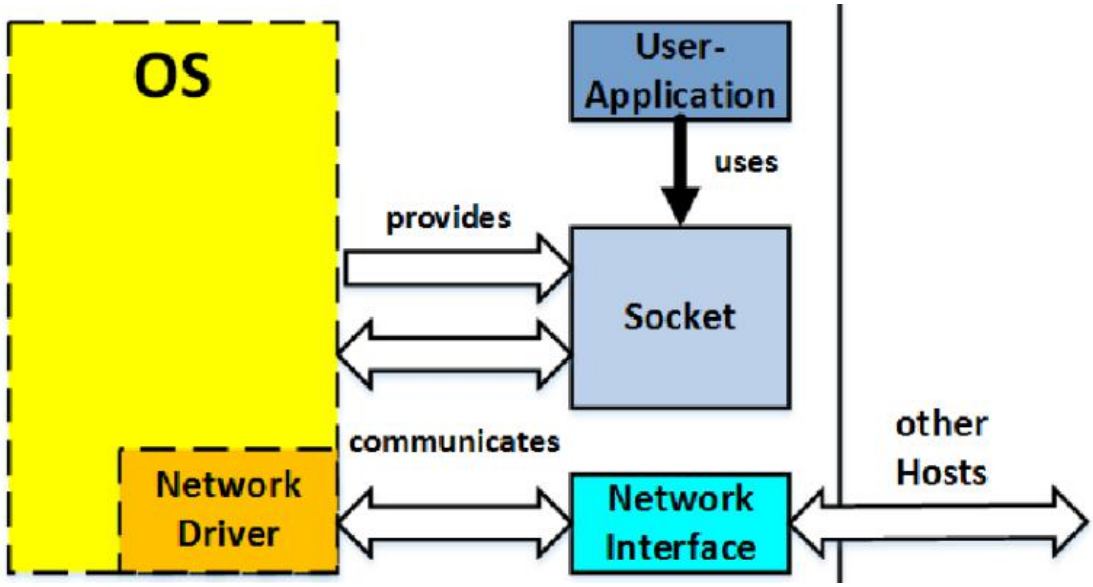
RAW SOCKETS

Name - Aman yadav
Roll - 171210007 (cse 3rd year)



A raw socket is a type of socket that allows access to the underlying transport provider. This topic focuses only on raw sockets and the IPv4 and IPv6 protocols. This is because most other protocols with the exception of ATM do not support raw sockets. To use raw sockets, an application needs to have detailed information on the underlying protocol being used.

There are two basic types of such **raw sockets**:

- The first type uses a known protocol type written in the IP header that is recognized by a Winsock service provider. An example of the first type of

socket is a socket for the ICMP protocol (IP protocol type = 1) or the ICMPv6 protocol (IP procotol type = 58).

- The second type allows any protocol type to be specified. An example of the second type would be an experimental protocol that is not directly supported by the Winsock service provider such as the Stream Control Transmission Protocol (SCTP).

Creating a Raw Socket

To create a socket of type **SOCK_RAW**, call the Socket function with the *af* parameter (address family) set to AF_INET or AF_INET6, the *type* parameter set to **SOCK_RAW**, and the *protocol* parameter set to the protocol number required. The *protocol* parameter becomes the protocol value in the IP header (SCTP is 132, for example).

Raw sockets offer the capability to manipulate the underlying transport, so they can be used for malicious purposes that pose a security threat.

Send and Receive Operations

Once an application creates a socket of type SOCK_RAW, this socket may be used to send and receive data. All packets sent or received on a socket of type SOCK_RAW are treated as datagrams on an unconnected socket.

The following rules apply to the operations over **SOCK_RAW** sockets:

- The sendto or WSASendTo function is normally used to send data on a socket of type **SOCK_RAW**. The destination address can be any valid address in the socket's address family, including a broadcast or multicast address. To send to a broadcast address, an application must have used setsockopt with SO_BROADCAST enabled.
  Otherwise, **sendto** or **WSASendTo** will fail with the error code WSAEACCES. For IP, an application can send to any multicast address (without becoming a group member).

- When sending IPv4 data, an application has a choice on whether to specify the IPv4 header at the front of the outgoing datagram for the packet. If the **IP_HDRINCL** socket option is set to true for an IPv4 socket (address family of AF_INET), the application must supply the IPv4 header in the outgoing data for send operations. If this socket option is false (the default setting), then the IPv4 header should not be in included the outgoing data for send operations.

- When sending IPv6 data, an application has a choice on whether to specify the IPv6 header at the front of the outgoing datagram for the packet. If the **IPV6_HDRINCL** socket option is set to true for an IPv6 socket (address family of AF_INET6), the application must supply the IPv6 header in the outgoing data for send operations. The default setting for this option is false. If this socket option is false (the default setting), then the IPv6 header should not be included in the outgoing data for send operations. For IPv6, there should be no need to include the IPv6 header. If information is available using socket functions, then the IPv6 header should not be included to avoid compatibility problems in the future. These issues are discussed in RFC 3542 published by the IETF. Using the **IPV6_HDRINCL** socket option is not recommended and may be deprecated in future.

- The recvfrom function is normally used to receive data on a socket of type **SOCK_RAW**. Both of these functions have an option to return the source IP address where the packet was sent from. The received data is a datagram from an unconnected socket.

Common Uses of Raw Sockets

One common use of raw sockets are troubleshooting applications that need to examine IP packets and headers in detail. For example, a raw socket can be used with the SIO_RCVALL IOCTL to enable a socket to receive all IPv4 or IPv6 packets passing through a network interface.

Limitations on Raw Sockets

On Windows 7, Windows Vista, Windows XP with Service Pack 2 (SP2), and Windows XP with Service Pack 3 (SP3), the ability to send traffic over raw sockets has been restricted in several ways:

- TCP data cannot be sent over raw sockets.

- UDP datagrams with an invalid source address cannot be sent over raw sockets. The IP source address for any outgoing UDP datagram must exist on a network interface or the datagram is dropped. This change was made to limit the ability of malicious code to create distributed denial-of-service attacks and limits the ability to send spoofed packets (TCP/IP packets with a forged source IP address).

- A call to the bind function with a raw socket for the IPPROTO_TCP protocol is not allowed.