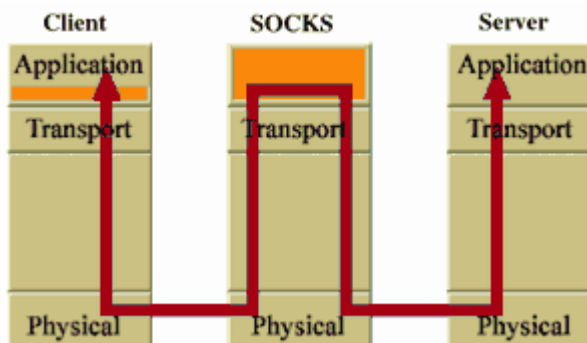


## SOCKS Overview

SOCKSv5 is an IETF (Internet Engineering Task Force) approved standard (RFC 1928) generic, proxy protocol for TCP/IP-based networking applications. The SOCKS protocol provides a flexible framework for developing secure communications by easily integrating other security technologies.

SOCKS includes two components, the SOCKS server and the SOCKS client. The SOCKS server is implemented at the application layer, while the SOCKS client is implemented between the application and transport layers. The basic purpose of the protocol is to enable hosts on one side of a SOCKS server to gain access to hosts on the other side of a SOCKS Server, without requiring direct IP-reachability.

## Place in OSI Layer



## What is a SOCKS Proxy Server?

When an application client needs to connect to an application server, the client connects to a SOCKS proxy server. The proxy server connects to the application server on behalf of the client, and relays data between the client and the application server. For the application server, the proxy server is the client.

## SOCKS Model

There are two versions of the SOCKS protocol - [SOCKSv4](#) and [SOCKSv5](#), respectively.

The SOCKSv4 protocol performs three functions:

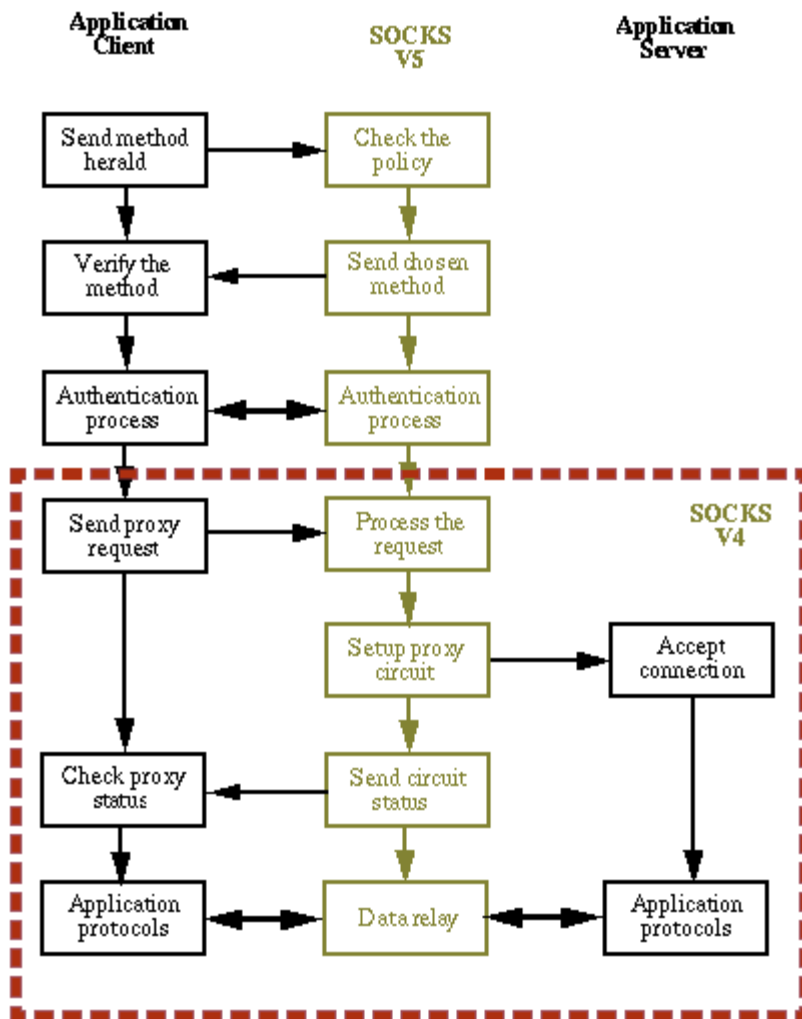
- makes connection requests
- sets up proxy circuits
- relays application data

The SOCKSv5 protocol adds authentication to the above functions.

## Control Flow of SOCKS

This figure shows the SOCKSv5 control flow model. The portion within the dashed-line represents SOCKSv4 functionality. Note that SOCKSv5 adds

authentication.



## Why SOCKS?

Because of its simplicity and flexibility, SOCKS has been used as a network firewall, generic application proxy, in virtual private networks (VPN), and for extranet applications. SOCKSv5-based applications offer many advantages due to its strong, yet flexible protocol framework:

- Transparent network access across multiple proxy servers
- Easy deployment of authentication and encryption methods
- Rapid deployment of new network applications
- Simple network security policy management

## Unique Features and Benefits with SOCKS

**A single communication protocol authenticates users and establishes the communication channel**

For each TCP or UDP communication channel that the SOCKS protocol

establishes, it:

- transfers user information from the SOCKS client to the SOCKS server for user authentication
- authenticates the user and the channel, and
- guarantees the integrity of TCP and UDP channels

Most tunneling protocols separate the authentication process and communication channel establishment, making it difficult to guarantee the integrity of the channels with authenticated users after multiple channels are established.

### **Application-Independent Proxy**

As a generic proxy, the SOCKS protocol establishes communication channels, and manages and protects the channel for any application. As new applications come to market, SOCKS can protect them without requiring additional development. IP layer stateful inspection proxies require a new script for protocol inspection, and application layer proxies require new proxy software for each new application.

### **Flexible protection through a variety of access control policies**

IP routers deliver IP packets by routing packets at the IP layer. Since SOCKS delivers TCP and UDP connections through a proxy mechanism at the TCP/UDP layer, it works with any application, and virtually all IP layer technologies, such as firewalls, NAT, and private IP. SOCKS adds the flexibility to manage the network through access control policies based on user, application, and time, in addition to source and destination addresses.

### **Bi-directional proxy support**

Most IP layer-based proxy mechanisms, such as network address translation (NAT), only support uni-directional proxy, from the internal (private IP) network to external network (the Internet). The proxy establishes the communication channel by manipulating IP addresses, therefore, the IP addresses must be routable on the Internet. These proxy mechanisms prevent applications (i.e. multimedia and collaborative applications) from establishing required return data channels (from the Internet to the intranet). In addition, IP layer-based proxy mechanisms need additional software modules for each application that uses multiple channels. SOCKS identifies communication targets through domain names, overcoming the private IP address restrictions. SOCKS can also use domain names to establish communication between separate LANs with redundant IP addresses.

## **SOCKS v4**

The [SOCKSv4 protocol](#) defines the message format and conventions to allow TCP-based application users transparent access across a firewall. During proxy connection setup, the SOCKS server grants access based on TCP header information including IP addresses, and source and destination host port numbers. The SOCKS server also authorizes users using ident (rfc1413) information.

The SOCKS user community proposed and implemented a [protocol extension](#) to SOCKSv4 that eliminates the requirement for SOCKSv4 clients to resolve internal and external domain names. By appending the unresolved domain names to the SOCKSv4 client requests, SOCKSv4 servers can attempt to resolve domain names.

Because of its simplicity, SOCKSv4 is widely used as a network firewall. There are two major weaknesses in SOCKSv4 protocol: lack of strong authentication and the requirement to recompile applications with SOCKSv4 client library. An IETF (Internet Engineering Task Force) working group drafted and approved a new version of SOCKS, [SOCKSv5](#). The working group completed three SOCKSv5-related standards: rfc1928, rfc1929, and rfc1961.

## SOCKSv4 Implementations

Networking Systems Laboratory (NWSL), formerly C&C Software Technology Center (CSTC) of NEC USA Inc., developed the most popular publicly available implementation. It includes the SOCKSv4 server and socksified versions of finger, whois, ftp, and telnet.

### SOCKS v5

The SOCKSv5 protocol, also known as authenticated firewall traversal (AFT), is an open Internet standard ([rfc1928](#)) for performing network proxies at the transport layer. It resolves several issues that SOCKS version 4 protocol did not fully address or omit:

- Strong authentication
- Authentication method negotiation
- Address resolution proxy
- Proxy for UDP-based applications

There are two additional SOCKSv5-related standards to support authentication methods:

- Username/Password authentication for SOCKSv5 ([rfc1929](#))
- GSS-API (Generic Security Service Application Programming Interface) authentication for SOCKSv5 ([rfc1961](#)).

## Authentication Method Negotiation

1. The application client declares to the SOCKSv5 server the authentication methods it can support
2. The SOCKSv5 server sends a message to the client announcing the method the client should use
3. The SOCKSv5 server determines the authentication method based on the security policy defined in the SOCKSv5 server's configuration. If the client's declared methods fail to meet the security requirement, the SOCKSv5 server drops communication.

## Address Resolution Proxy

SOCKSv5's built-in address resolution proxy simplifies DNS administration and facilitates IP address hiding and translation. SOCKSv5 clients can pass the name,

instead of the resolved address, to the SOCKSv5 server and the server resolves the address for the client.

## Proxy for UDP-based Applications

SOCKSv5 supports **UDP association**. UDP association creates a virtual proxy circuit for traversing UDP-based application data. There are two differences in TCP and UDP-based proxy circuits:

- The proxy circuit for UDP is a pair of addresses for the communication endpoints that send and receive datagrams.
- UDP proxy headers encapsulate application data, including the destination address of a datagram.

## SOCKSv5 Implementations

For Permeo's reference implementation of the SOCKSv5 protocol, see [SOCKSv5 Reference Implementation](#).

For additional security and connectivity features, see Permeo's commercial line of products - [www.permeo.com](http://www.permeo.com)

Dieses Dokument wurde mit Win2PDF, erhaeltlich unter <http://www.win2pdf.com/ch>  
Die unregistrierte Version von Win2PDF darf nur zu nicht-kommerziellen Zwecken und zur Evaluation eingesetzt werden.