EXHIBIT E3

DECLARATION OF JAMES CHESTER

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Patent No. 7,490,151	)		
	)		
Filed: September 30, 2002	)	Group Art Unit:	Central Reexamination Unit
	)		
Issued: February 10, 2009	)		
	)	Examiner:	
Inventors: Munger et al.	)		
	)	Confirmation No.:	
For: ESTABLISHMENT OF A SECURE COMMUNICATION LINK BASED ON A DOMAIN NAME SERVICE (DNS) REQUEST	)		

DECLARATION OF JAMES CHESTER UNDER 37 C.F.R. § 1.132

I, JAMES SAMUEL CHESTER, do hereby declare and state:

1. I am a citizen of the United States, and reside in Florida. My c.v. is attached as Exhibit A.

2. I am being compensated for my time at a rate of $375.00 per hour.

3. In addition to the documents provided as exhibits to this declaration, I have reviewed documents including the following:

   - U.S. Patent No. 7,490,151 (the '151 patent);

   - Declaration of Jason Nieh from Reexamination Control No. 95/001,269.

A. My Background

4. I am presently CEO of a software development and consulting firm called Assured Products Group, which specializes in software development, consulting, and regulatory compliance.

5. From March 1992 to August 2002, I was employed by the International Business Machines Corp. (IBM). During the period 1996 to 2002, I was responsible for global strategic initiatives overseeing design and implementation of secure networking services, architecture, and cost reductions for IBM worldwide and IBM clients. In that role, I evaluated network security products and services from many vendors, and for designing and implementing these products and services that IBM designed and implemented for its clients.

6. Between 1996 and 2000, I recall receiving a number of VPN networking products from Aventail, Inc. I recall using these Aventail VPN products to develop virtual private networking solutions for several hundred IBM clients during this period as well as remote access systems used by IBM employees worldwide. CSC, DuPont, and a number of companies had deployed the Aventail solutions and I gave many seminars during this period describing secure communication designs that used the Aventail solutions. Competitors quickly adopted the virtual secure network and communication architecture we employed with Aventail.

7. IBM was advancing the use of both hosted and distributed systems through secure networks to routinely communicate company private information internally as well as externally for mobile computing employees and employees assigned behind firewalls on customer premises. I estimate that solutions based on Aventail products were deployed to more than 65,000 users in IBM alone by the end of 1998, and were deployed to many thousands more during 1999.

8. In my role as Vice President or Strategy and Strategic Initiatives, I oversaw and operationally deployed networking security solutions that leveraged the improvements that we saw in the 1990s in the core elements of networking security; namely, communications, routing, security, verification, and paths. In that role, I led activities for internal and external network design, development, and solutions. That included all aspects from access to verification to exchange.

B. Aventail VPN Products Distributed Between 1996 and 2000

9. Aventail distributed several VPN products during the period 1996 and 2000. Each of these products included a server component and a client component.

10. One Aventail VPN solution included client software called AutoSOCKS which could be paired with two versions of VPN server software. One version of the Aventail server software was focused on remote employees and was called MobileVPN. The other package focused on non-employees and was called PartnerVPN. Both server products functioned identically in how they worked with the AutoSOCKS client to automatically establish VPNs between remote users and private network resources.

11. Aventail distributed several versions of AutoSOCKS and MobileVPN/PartnerVPN products between 1996 and 2000. Each new version of each component had a higher version number.

12. The Aventail products were distributed with installation discs and printed manuals for each of the software packages. Exhibit B is a copy of the Administrator's Guide for version 2.1 of the Aventail AutoSOCKS client software. I received this document on approximately December 1997.

13. At the time I received Exhibit B, I was under no obligation to keep this document secret or to not distribute it to others. In fact, we distributed the AutoSOCKS Administrator's Guide with the other printed materials that came with the Aventail AutoSOCKS/VPN Server to IBM clients to whom we deployed VPN solutions, as well as IBM employees.

By the spring of 1998 we were giving seminars and interviews on the solution and benefits.

14. A second VPN solution Aventail distributed between 1996 and 2000 was called the Aventail Extranet Center (AEC). This product included client software called Aventail Connect and server software called Aventail Extranet Server.

15. I recall that Aventail announced its AEC v3.0 product in the fall of 1998, and began distributing this product no later than mid-January of 1999. Because IBM was the largest user of Aventail VPN products, we would be one of the first companies to receive new versions of the Aventail products; both evaluation and production products. I was personally involved in Aventail's strategic planning and direction from March 1998.

16. The AEC v3.0 product included version 3.01/2.51 of the Aventail Connect software, and version 3.0 of the Aventail Extranet Server.

17. Exhibit C is a copy of the Administrator's Guide for Aventail Connect v3.01/2.51. I recall receiving Exhibit C with the AEC v3.0 product no later than July 1998.

18. At the time I received Exhibit C, I was under no obligation to keep this document secret or to not distribute it to others. Like earlier Aventail products, we distributed copies of the AutoSOCKS Administrator's Guide along the other printed materials that came with the Aventail AutoSOCKS/VPN Server to IBM clients to whom we deployed VPN solutions, and to IBM employees using the Aventail Connect v3.01/v2.51 client.

19. The AEC product was a very versatile and stable VPN solution. It received very good reviews from the technical press. We deployed VPN solutions based on this product to more than 20,000 IBM employees domestically by March 1998 and more than 65,000 IBM employees worldwide by July 1998.

20. Aventail distributed an updated version of the AEC product in the summer of 1999. This updated version was designated AEC v3.1, and included Aventail Connect v3.1/2.6 and Aventail Extranet Server v3.1.

21. I recall receiving the AEC v3.1 product no later than the end of June of 1999. The product I received included the installation discs for the Aventail Connect v3.1/2.6 client software and v3.1 of the Aventail Extranet Server software. It also included printed manuals for these products, including the Aventail Connect v3.1/2.6 Administrator's Guide, a copy of which is shown in Exhibit D.

22. At the time I received Exhibit D, I was under no obligation to keep this document secret or to not distribute it to others. Again, as was the case with earlier Aventail products, we distributed copies of the AutoSOCKS Administrator's Guide along the other printed materials that came with the Aventail AutoSOCKS/VPN Server to IBM clients to whom we deployed VPN solutions, and to IBM employees that were using the Aventail Connect v3.1/2.6 client. By the summer of 1999, this product was routinely packaged with IBM offerings in all business sectors worldwide. Competitors were deploying this product as well in the same timeframe.

C. Relevant Background on TCP/IP Communications

23. Two of the claims of the '151 patent refer to IP hopping schemes or regimes. I have been asked to provide some background on IP hopping schemes.

24. The TCP/IP protocol was designed to employ flexible routing of traffic. IP packets are datagrams that contains source and destination IP addresses. These IP packets or datagrams traverse a network by being routed between devices (also called nodes). Under the design of the TCP/IP protocol, there is no requirement for an IP packet to take a pre-defined path from source to destination unless the IP packet is being routed over a statically configured network. IP packets thus can take multiple different paths to reach the same destination.

25. Routes that an IP packet will take from source to destination are determined by each node/host as it processes the IP datagram. The destination IP of the datagram is compared to a locally maintained routing table, and the IP packet is forwarded as deemed appropriate.

26. TCP/IP hosts use a routing table to maintain knowledge about other IP networks and IP hosts. Networks and hosts are identified by using an IP address and a subnet mask. In addition, routing tables are important because they provide needed information to each local host regarding how to communicate with remote networks and hosts. Because it is impractical for each computer on an IP network to maintain a routing table having entries for every other computer or network that communicates with it, a default gateway (IP router) is used instead.

27. When a computer prepares to send an IP datagram, it inserts its own source IP address and the destination IP address of the recipient into the IP header. The computer then examines the destination IP address, compares it to a locally maintained IP routing table, and takes appropriate action based on what it finds. The computer does one of three things:

    a. It passes the datagram up to a protocol layer above IP on the local host.

    b. It forwards the datagram through one of its attached network interfaces.

    c. It discards the datagram.

28. When the computer searches the routing table to identify a route for an IP packet, it will look for the closest match to the destination address. The most specific to the least specific route is searched for in the following order:

    a. A route that matches the destination IP address (host route).

    b. A route that matches the network ID of the destination IP address (network route).

c. The default route.

29. If a matching route is not found, the datagram is discarded.

30. Routing algorithms can be static or dynamic. A static route implies that every route is known and manually entered. Dynamic routing uses tables that are updated via different protocols. The two most commonly used protocols in the 1997-2000 time frame were the RIP protocol (see, Malkin, G., "RIP Version 2," RFC 2453 (November 1998) (available at http://www.networksorcery.com/enp/rfc/rfc2453.txt)) and the OPSF protocol (see Moy, J., "OSPF version 2," RFC 2328 (April 1998) (available at http ://tools.ietf.org/html/rfc2328).

31. Communications between networks relies upon traffic being routed from the source to the destination. In TCP/IP communications, the source will encapsulate a TCP message within an IP datagram; the header of the datagram will contain the source and destination addresses. The source will then encapsulate the IP datagram into a layer 2 frame for transmission across the internetwork hop/link. IF the source does not know the route to the destination address, it will send the packet to its network gateway. From the gateway until the destination, each node will remove the IP datagram from its layer 2 encapsulation. The destination address contained within the IP datagram header is then compared to the current node's routing table and the best route is determined. The node will then re-encapsulate the IP datagram into a layer 2 frame and pass the datagram along the next hop/link of the network that best matches the destination address.

32. Dynamic routing based on RIP or OPSF standards are inherently flexible. So, if one link the network goes down or becomes unavailable, the route can be changed. Routing tables are simply updated to provide the new routes, and the packets get sent along a different path to the destination. When IP packets are sent over the public Internet, and are not routed manually, they inherently will follow pseudorandom paths – the path is not defined until its actually taken by the IP packet, and each IP packet will likely travel on different paths even when the source and destinations are the same.

33. Any TCP/IP communication will inherently meet the requirement in the '151 patent claims that IP packets must be sent from a client computer to a secure destination by an "IP hopping" scheme or regime. As I explained above, IP hopping is integral to the design of TCP/IP communications, and occurs whenever an IP packet is sent from a source to a destination via a TCP/IP communication.

D. Observations of the Declaration of Dr. Jason Nieh in a Prior Reexamination

34. I reviewed a declaration submitted by Dr. Jason Nieh in Reexamination Control No. 95/001,269 involving U.S. Patent No. 6,502,135, the parent of the '151 patent. I believe several of Dr. Nieh's statements in his declaration do not accurately describe how the Aventail products work or what is described in the Administrator's Guide for Aventail Connect v3.1/2.6 (Exhibit D).

35. Dr. Nieh's general conclusion is that the Aventail VPN products did not establish VPNs as they are defined in the claims of the '135 patent. This is incorrect. Based on my

personal experience using these products, they were routinely used by remote users to establish VPNs, and that those VPNs met the requirements of the claims of the '135 patent.

36. Initially, I understand that a court has interpreted certain phrases in the claims of the '135 patent, including "virtual private network" or "VPN." The meaning of VPN was simply "a network of computers which privately communicate with each other by encrypting traffic on insecure communication paths between the computers." This is precisely what Exhibit D shows client computers running Aventail Connect doing.

37. Initially, Dr. Nieh states that he believes the Aventail Connect v3.1/2.6 client software was simply a SOCKS v5 client. See Nieh Declaration, paragraphs 11 to 14. Aventail Connect v3.1, like the earlier Aventail Connect v3.01/2.51 and the Aventail AutoSOCKS clients, did much more than handle SOCKS transactions.

38. The Aventail Connect and Extranet Server, like the earlier Aventail VPN solutions, used well-established network security techniques. For example, the Reverse Address Resolution Protocol (RARP) has been in existence since June 1984, while session IDs and session synchronization have been in existence since the 1960s.

39. Client computers running each of the Aventail clients (i.e., Aventail Connect and Aventail AutoSOCKS) could automatically establish VPNs that allowed a remote user to gain access to secure resources on a private network. These products all worked by transparently intercepting and evaluating DNS and TCP/IP connection requests made on the client computer.

40. The Aventail clients could be configured in one of two ways. First, they could be configured to determine if a connection request specified a destination that required a VPN (e.g., a secure website on a private network). If so, the client would automatically re-route that connection to a VPN server, manage authentication of the user with the VPN server, and encrypt/decrypt the outgoing and incoming network traffic.

41. Second, the Aventail clients could be configured to route all DNS requests containing hostnames that could not be resolved on the local computer to a VPN server. In this configuration, the client computer would establish a connection to the VPN server, authenticate itself with the server, and if that was successful, it would then send the hostname from the original DNS request to the VPN server. The VPN server (i.e., Aventail Extranet Center or Aventail MobileVPN/PartnerVPN) would then resolve the hostname (if necessary) and then determine if a VPN was needed between the requesting client and the destination. If no VPN was required, the VPN server would return the resolved IP address back to the client.

42. In either configuration, if a connection request was not seeking access to a secure destination requiring a VPN, it would just be handed off to the normal TCP/IP handling procedures of the client computer for handling. So, for example, if the VPN server had done the hostname resolution and returned a resolved IP address, WinSock and the

TCP/IP stack on the client computer would then just establish a connection to the specified IP address without the Aventail client being involved any further.

43. For the Aventail Connect v3.1/2.6 client, this functionality is described on Page 9 of Exhibit D:

> When the Aventail Connect LSP receives a connection request, it determines whether or not the connection needs to be redirected (to an Aventail ExtraNet Server) and/or encrypted (in SSL). When redirection and encryption are not necessary, Aventail Connect simply passes the connection request, and any subsequent transmitted data, to the TCP/IP stack.

44. Exhibit D describes a VPN design made up of client computers running Aventail Connect v3.1/2.6 connecting to a private network through an Aventail VPN server. See Exhibit D at pages 76 to 78. I note that Dr. Nieh did not discuss this section of Exhibit D in his declaration.

45. Page 77 of Exhibit D explains this VPN implementation as follows:

> The mobile user workstations connected to the public Internet are the client workstations, onto which, Aventail Connect will be deployed. Due to the routing restrictions described above, these clients will have no network access beyond the Aventail ExtraNet Server unless they are running Aventail Connect. Depending on the security policy and the Aventail ExtraNet Server configuration, Aventail Connect will automatically proxy their allowed application traffic into the private network. In this situation, Aventail Connect will forward traffic destined for the private internal network to the Aventail ExtraNet Server. Then, based on the security policy, the Aventail ExtraNet Server will proxy mobile user traffic into the private network but only to those resources allowed. The client workstations we focus on in this section are Microsoft Windows based PCs. (emphasis added)

46. Exhibit D on pages 12 to 13 also explains that client computers running Aventail Connect will automatically handle authentication of the remote user, as well as encryption of the traffic between the remote user's computer and the private network:

> User authentication and encryption on the Aventail ExtraNet Server require all users to use Aventail Connect to authenticate and encrypt their sessions before any connection to the internal private network(s). For this example, the Aventail ExtraNet Server encrypts all sessions with SSL.

47. All of this functionality was also present in Aventail Connect v3.01/2.6 and AutoSOCKS v2.1. See Exhibit B at pages 7 to 9 and 37 to 39; Exhibit C at 11 to 12 and 72 to 74.

48. Dr. Nieh provides three general reasons why he believes that Aventail Connect and Aventail Extranet Server did not create VPNs within the meaning of the '135 patent claims.

49. First, in paragraph 20, Dr. Nieh states:

> Aventail has not been shown to demonstrate that computers connected via the Aventail system are able to communicate with each other as though they were on the same network. Aventail discloses establishing a point-to-point SOCKS connection between a client computer and a SOCKS server. According to Aventail, the SOCKS server then relays data received to the intended target. Aventail does not disclose a VPN, where data can be addressed to one or more different computers across the network, regardless of the location of the computer.

50. I found nothing in the claims of the '135 patent that require that a computer connected to a private network to communicate directly with other remote computers that are also connected to the network. Instead, all that is required for there to be a VPN is "a network of computers which privately communicate with each other by encrypting traffic on insecure communication paths between the computers.."

51. Regardless, Exhibit D shows that client computers running Aventail Connect v3.1/2.6 did have the ability to communicate with other computers on the network, including other remote users.

52. For example, the figure on page 77 of Exhibit D shows a remote user using a client computer running Aventail Connect to gain access to and to interact with different computers on a private network. Network traffic from the remote computers is sent into the private network, and handled by the rules and policies governing all network traffic on that private network. As explained on page 77 of Exhibit D:

> Depending on the security policy and the Aventail ExtraNet Server configuration, Aventail Connect will automatically proxy their [the client computer's] allowed application traffic into the private network. In this situation, Aventail Connect will forward traffic destined for the private internal network to the Aventail ExtraNet Server. Then, based on the security policy, the Aventail ExtraNet Server will proxy mobile user traffic into the private network but only to those resources allowed.

53. This explanation makes it clear that a remote user running Aventail Connect could interact with any resources on the private network that the user was authorized to access. It also shows that a mobile user connected through Aventail Connect will be equivalent to a local user on the private network – both the remote and local users will be able to send and receive traffic to and from destinations on the private network. So, if network policies allowed a user to route network traffic to other users on the private network (including remote users), a remote user connected to that network through the Aventail VPN solution will be able to communicate to those other users.

54. Exhibit D also describes the ability of client computers running Aventail Connect to dynamically navigate resources made available on a private network via the "Secure Extranet Explorer" capability of Aventail Connect. As Exhibit D explains on page 95:

> Secure Extranet Explorer (SEE) allows you to view your Extranet Neighborhood, which is accessed through the Extranet Neighborhood icon on your desktop. The Extranet Neighborhood user interface resembles that of Network Neighborhood. However, while Network Neighborhood displays all computers on your local network, Extranet Neighborhood allows you to browse, copy, move, and delete files from remote computers via the Aventail Connect extranet connection. With Extranet Neighborhood, all interaction with the remote server can be secured. Network administrators determine which local and remote computers are available to users.

55. So, if a private network had the appropriate policies in place, a remote user would be able to view and access resources on any of the computers on the network, including those on other remote computers/servers.

56. The second reason Dr. Nieh offers to support his belief that the Aventail VPN products did not establish VPNs was that "Aventail Connect's fundamental operation is incompatible with users attempting to transmit data that is sensitive to network information." See Nieh Declaration at paragraph 24-25.

57. Dr. Nieh incorrectly suggests that applications making DNS requests will try to make connections using the "false network information" that Aventail Connect uses to flag connection requests requiring a VPN. Specifically, Dr. Nieh says that:

> 24. Because Aventail discloses that Aventail Connect operates between these layers, Aventail Connect can intercept DNS requests requested by the user. Aventail Connect intercepts certain DNS requests, and returns a false DNS response to the user if the requested hostname matches a hostname on a user-defined list. Accordingly, Aventail discloses that the user will receive false network information from Aventail Connect for these hostnames.
>
> 25. If the client computer hopes to transfer to the target data that is sensitive to network information, this falsification of network information would prevent the correct transfer of data. A client and target connected according to Aventail would be unable to transfer data as they otherwise would have been had they been on the same network. Thus, Aventail has not been shown to disclose a VPN

58. Dr. Nieh has apparently misunderstood how client computers running Aventail Connect actually function.

59. As explained on pages 11 to 13 of Exhibit D, Aventail Connect would monitor DNS requests made by applications on the client computer. If the DNS request contained a

hostname instead of a literal IP address, and that hostname specified a secure destination that required a VPN, then Aventail Connect would insert a false hostname into the DNS request. See, e.g., Exhibit D at page 12 ("If the destination hostname matches a redirection rule domain name (i.e., the host is part of a domain we are proxying traffic to) then Aventail Connect creates a false DNS entry (HOSTENT) that it can recognize during the connection request.") Similarly, if the Aventail Connect client computer was configured to proxy all non-local DNS requests to a separate computer for resolution, Aventail Connect would return a false DNS entry that it could recognize later to the requesting application.

60. In a second step, Aventail Connect would evaluate the connection request to see if it had to be redirected to the VPN proxy server. If a connection request contained the "false" information inserted in step 1 or because the real IP address was on a redirection list[1], the Aventail Connect client would establish a connection to the VPN proxy server (i.e., Aventail Extranet Server). The VPN Server and the Aventail Connect Client would then perform authentication, and if that was successful, the VPN would be established between the client computer and the destination computer specified in the original connection request. See, Exhibit D, pages 12-13 (steps 2 and 3). False hostnames thus were used by Aventail Connect to flag DNS requests requiring redirection to a VPN server during the TCP connection process, and would not cause client computers to attempt to connect to "false" destinations.

61. As Exhibit D explains on pages 12-13, this entire process is transparent to the client computer: "From the application's point of view, the entire SOCKS negotiation including the authentication negotiation, is merely the TCP handshaking." In other words, the application on the client computer requesting a connection would not act on the "false hostname" information, but would simply see a connection being established with the destination specified in the connection request.

62. I also recall based on my personal experiences using Aventail Connect v3.1/2.6 (and with earlier versions dating back to 1997) that client computers running Aventail Connect were able to transfer data to a private network as if they had been on the same network, and that the false hostname inserted in DNS requests by Aventail Connect did not impede or disrupt the secure communications between the client and the private network, specifically useful with large host applications, including expense reports, technical journals, images, program specific communications and operations management.

63. Dr. Nieh's third reason why he believes that Aventail Connect did not establish VPNs is that he believes computers running Aventail Connect "do not communicate directly with each other." This is incorrect. As explained above, Exhibit D shows traffic from a client computer running Aventail Connect being automatically proxied into the private network. What the network does with that traffic at that point was dictated by the policies that were enforced on the network. If those policies permitted a user to interact with another

[1] As explained in step 2(a) on page 12 of Exhibit D, if the DNS request did not include a host name, but was a real IP address (e.g., 1.2.3.4), then no DNS resolution step is needed.

computer attached to the private network through Aventail Connect, then that traffic would be routed to that computer. This capability was particularly useful for employees at customer locations behind external firewalls.

64. In addition, as I explained in paragraphs 50 to 54 above, the Secure Extranet Explorer functionality of Aventail Connect described on pages 95 to 106 of Exhibit D enabled remote users running Aventail Connect on client computers to see, navigate to and access resources on other remote computers.

65. If Dr. Nieh's belief is that a client computer could not establish a VPN if it did not send its network traffic "directly" to another computer (i.e., not via a gateway computer or other intermediary computer), then no VPN could ever be established. Any form of network traffic is inherently routed over intermediary nodes; this is a central feature of the TCP/IP protocol. A client computer does not have to establish a direct connection to a destination computer in order to establish a secure connection to a destination computer; the fact that its traffic will pass through an intermediary computer, such as the Aventail VPN server as described on pages 76 to 78 of Exhibit D, is simply irrelevant.

66. In the Aventail VPN solution – like other types of VPN solutions – an intermediary computer (e.g., a proxy server or gateway) evaluates incoming traffic, blocks unauthorized traffic, and regulates authentication, encryption and transit of the incoming and outgoing traffic. The communication between a remote user on a client computer occurs via the intermediary computer (e.g., the VPN server) to the destination on the private network.

67. Dr. Nieh apparently has confused the issue of network communication with the issue of how network traffic is routed. Exhibit D plainly shows client computers running Aventail Connect communicating privately with destination computers on the private network via insecure communication paths (i.e., over public networks, such as the Internet). These communications are encrypted, and enable the client computer to gain access to resources on a private network.

68. Dr. Nieh also expresses his opinion that Exhibit D does not disclose a virtual private network according to claim 10 of the '135 patent. This claim defines a VPN system. Dr. Nieh does not include any particular reasons to support his conclusion, other than to refer to his other opinions in the declaration. See Nieh Declaration at Paragraphs 28 and 29.

69. Finally, Dr. Nieh inaccurately discusses DNS resolution on client computers running Aventail Connect software. In paragraph 30, he describes requirements listed in claim 10 of the '135 patent. One of these is that "a DNS proxy ...returns the IP address for the requested domain name if it is determined that access to a non-secure web site has been requested." Dr. Nieh then states in paragraph 32 that Aventail Connect will not do this because "if the hostname matches a local domain string or does not match a redirection rule, Aventail Connect passes the name resolution query through to the TCP/IP stack on the local workstation. The TCP/IP stack performs the lookup as if Aventail Connect were not running."

70. Dr. Nieh appears to be seeing the trees but not the forest in this case. What is being described on pages 11 to 14 of Exhibit D is a client computer that us running both the Windows operating system and Aventail Connect. If the client computer configured with both the Windows operating system software and Aventail Connect determines that a DNS request is not requesting access to a secure website requiring a VPN, the client computer will "return the IP address for the requested domain." It will do this by the ordinary DNS resolution capabilities of WinSock and the TCP/IP stack within Windows as described on page 8 of Exhibit D (i.e., "the application executes a Domain Name System (DNS) lookup to convert the hostname into an Internet Protocol Address.") Thus, the client computer running Windows will perform this step listed in claim 10 of the '135 patent.

71. In addition, page 45 of Exhibit D shows that Aventail Connect could be configured "to resolve hostnames locally without needing to venture on to the Internet." Importantly, this ability of Aventail Connect to resolve hostnames was not limited to hostnames requiring redirection to a VPN server. Instead, it was simply a DNS resolution capacity built into the Aventail Connect client.

72. Based on these observations, I do not agree with Dr. Nieh's conclusion that Aventail Connect did not meet the requirements of the claims of the '135 patent. Instead, based on my personal experience and daily operations with tens of thousands of users by the spring of 1998, I believe there is nothing new or innovative about the processes and systems defined by the claims of the '135 patent. Similarly, I do not believe there is anything new or innovative about the processes and systems defined by the claims of the '151 patent.

* * * * * * * * * * *

I declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the patent subject to this reexamination proceeding.

James Chester

25 Jul 11
Date

EXHIBIT A

CURRICULUM VITAE OF JAMES CHESTER

ASSURED PRODUCTS GROUP

James Chester

Mr. Chester is Chief Executive Officer of Assured Products Group, and is directly accountable and responsible for all aspects of the company. Assured Products Group helps customers build brand equity and improve revenue while addressing emerging economic, compliance, and supply chain challenges. The Trusted Trade Environment provides an integrated solution to create segment growth, brand loyalty, and recurring revenue while assuring supply chain integrity and compliance with domestic and international regulatory requirements. Assured sourcing provides assisted through turn key sourcing of compliant products from low cost manufacturing environments while assuring customer schedule, quality, and regulatory compliance. Investigation and enforcement services provide IPR/Brand protection, liaison with authorities, import/export, and supply chain assurance.

Mr. Chester provides executive leadership and personal involvement with engineering, development, and operations team members, while working closely with clients to ensure satisfaction and realization. He has been involved in global design and development decisions at numerous Fortune 50 companies. Mr. Chester establishes actionable transformation processes and solutions to help reduce cost without reducing staff. He is also personally involved in customized trade environment design and development for key customers worldwide. The company supports emerging security standards and development of near-term and long-term solutions for commerce, transportation, security, and law enforcement agencies. Some notable customers include U.S. federal and state agencies, Interpol, Boeing, DuPont, 3M, Dynic, Sempra Energy, Zurich Financial Services, Cisco Systems, PRYM Consumer, Mattel Corporation. We have ten-year agreements with Juvenile Product Manufacturers Association, Craft and Hobby Association, Game Manufacturing Association, and American Home Furnishings Association.

During his twenty five plus year career in the information systems, aerospace and utilities industries, Mr. Chester has held a number of senior executive and key corporate positions at IBM and major Aerospace corporations.

As IBM's Vice-President of Strategic Initiatives and Global Architecture, Mr. Chester led the strategic direction, prototype, and operational guidance of the IBM infrastructure worldwide. He led the identification and realization of more than $250 Million in year over year savings while accommodating an annual 6% worldwide growth. The resulting infrastructure supports more than 360,000 employees and contractors worldwide and forms the core infrastructure for IBM's e-business enablement and business transformation initiatives. Some highlights include the Global Web Architecture, Secure Inbound Network Environment, Information Warehouse Architecture, Global Security Architecture and Operations, Standardized Offerings, Alternative Access, and Global Ethernet Architecture and Migration, including integrated secure wireless access. Mr.

Chester also led the review and recommendation of emerging solutions to consolidate custom and redundant applications development environments. This responsibility included the evaluation of programming decisions including personnel mix and requirements for 9,000 developers. Mr. Chester led the assessment of legacy solutions and implementation of integrated supply chain and CRM applications worldwide. The promotion to production strategy allowed IBM to utilize numerous third party development partners for IBM.com, w3.IBM.com, and partner relations. He showcased and marketed IBM solutions and large account outsourcing and applications development across multiple industries. Mr. Chester was also a member of IBM's Architecture Board representing Global Services and Corporate Initiatives worldwide.

Mr. Chester was the senior IBM project executive for Northrop Grumman Corporation and The Boeing Company. He directed the provisioning of a distributed desktop environment and end-to-end computing solution for the Military Aircraft Systems Division at Northrop. During his tenure at The Boeing Company, Jim developed and led the enterprise systems management project that provided architecture and management processes for Boeing's distributed server architecture worldwide. This solution forms the basis for Boeing's information backbone for all divisions and operations, including manufacturing, distribution, and support. Mr. Chester was the senior executive for the Gulfstream Corporation leading the review and analysis of parallel aircraft manufacturing and quality engineering process improvements. This included a review and recommendation of emerging parts, manufacturing, and supply chain management for the GIV and GV aircraft. He also led strategic development and solutions for the Aerospace and Utilities Industry of IBM Global Services. Mr. Chester performed and participated in many executive reviews of technology and cost savings considerations for AXA, Dynergy, CISCO, Northern Telecom, McDonnell Douglas, Boeing, Northrop Grumman, United Technologies, Raytheon, Tenneco, El Paso Gas, Textron, and Sempra Energy.

Jim led the development and deployment of ground systems and satellite control for direct broadcast and data transmission in the Americas, Asia Pacific, Europe and Africa. Projects also included ground and spacecraft architectures for low, medium, and geostationary satellite telephony. This included design and joint development with Nokia, India Space Agency, Hughes Aircraft, and INMARSAT. He also directed weather and imagery exploitation programs for the collection, evaluation, and dissemination of atmospheric and surface information for civilian, military, and commercial use in Australia, Hong Kong, Singapore, the United Kingdom, and the United States.

Manned space program experience includes the US space shuttle for NASA and DoD, ground systems control, orbiter flight software and control, the European Space Agency Hermes Project, and Japan's space station module and safety, quality, and reliability programs. Arian launch and systems support including landing and failure analysis were key program responsibilities as well.

Mr. Chester led Independent Research and Development of neural networks, digital voice, space systems exploitation, heads-up displays, network integration, alternative access, and command and control systems. He also led the investigation and feasibility studies of emerging voice, video, and data integration of heterogeneous spacecraft environments.

Mr. Chester has United States patents in the application and development of Internet and intranet technologies and ownership and verification services. He also sponsored patents in wireless and wired computing technologies. He is a past member of California Edison's Strategic Supplier Technology Board, participating in the review and recommendation of solutions for the regulated and unregulated energy conglomerate.

Mr. Chester has been interviewed, quoted, and published in Network World, Information Week, Space Symposiums, European Space Agency, Japanese Space Agency, USA Today, and regional and local news and television and numerous Fortune 50 customers.

EXHIBIT B

AVENTAIL AUTOSOCKS V2.1 ADMINISTRATOR'S GUIDE

2.1

Aventail AutoSOCKS

ADMINISTRATION & USER'S GUIDE

AVENTAIL

Aventail AutoSOCKS v2.1 *Administration and User's Guide*

117 South Main Street
4th Floor
Seattle, WA 98104-2540
USA

Printed in the United States of America.

Trademarks and Copyrights

Table of Contents

Introduction

Welcome to the AutoSOCKS™ v2.1 secure Windows client for 16- and 32-bit Windows applications. AutoSOCKS v2.1 is the first commercial application to incorporate the SOCKS v5 security protocol standard, simplifying SOCKS deployment for end users and network managers.

AutoSOCKS transparently intercepts WinSock communication requests issued by TCP/IP applications and processes them based upon a set of routing directives (rules) assigned when AutoSOCKS is configured. (For more information about WinSock, TCP/IP, and general network communications, see "Getting Started.")

On larger networks, AutoSOCKS can address multiple SOCKS v5 servers based on end destination and type of service. This feature enables network administrators to effectively monitor and direct network traffic.

Features of AutoSOCKS v2.1:

- Supports both SOCKS v4 and SOCKS v5 standards
- Supports RFC1928 and RFC1929 SOCKS v5 standards
- Network-based setup provides a single configuration point with a simple user interface
- Transparently route connections from Windows applications to external networks through any SOCKS-based firewall system
- Logging utility to troubleshoot problems with network connections
- Enables internal network connections to pass through without interference
- Enables network redirection through multiple SOCKS servers
- Supports multiple authentication methods including SOCKS v4 Identification, username/password, CHAP, and SSL 3.0. Other authentication modules can be added
- Supports 16-bit WinSock 1.1 applications under Windows 3.1 and Windows for Workgroups 3.11
- Supports both 16- and 32-bit applications under Windows 95, Windows NT 3.51, and Windows NT 4.0
- Provides automated installation and uninstallation
- WinSock interoperability tested at Stardust WinSock Labs

About This Document

The AutoSOCKS v2.1 *Administration and User's Guide* provides basic information about AutoSOCKS v2.1. It is designed to include entry-level data for non-technical users as well as more advanced installation, setup, and configuration information for network administrators.

This information is also available via online AutoSOCKS Help and the Aventail web site at **http://www.aventail.com/**.

Document Organization

This document is divided into two primary sections: the Administrator's Guide and the AutoSOCKS *Utilities Reference Guide*. The Administrator's Guide describes procedures for setting up, installing, and configuring AutoSOCKS for individual and multiple networked workstations.

The AutoSOCKS *Utilities Reference Guide* describes the AutoSOCKS system menu commands and utility programs. It contains detailed information about using Ping and Traceroute utilities and documents the authentication/encryption modules and settings.

In addition to the AutoSOCKS v2.1 *Administration and User's Guide* and the AutoSOCKS *Utilities and Reference Guide*, this document includes a removable AutoSOCKS User's Supplement which describes screen displays and features that end-users may encounter while running AutoSOCKS in their client workstations. The document concludes with Appendix 1: Troubleshooting and a Glossary.

Check the Quick Start Card, a short document designed to help you install AutoSOCKS to an individual workstation.

Document Conventions

The following typographic conventions are used in this document. Exceptions may be made for online material; for instance, italics may be difficult to read online.

Convention	Usage
ALL CAPITALS	Filenames and extensions, directory names, keynames, and pathnames.
Bold	Anything the user types, including command-line commands, addresses or URLs, options, and portions of syntax that must be typed exactly as shown. Dialog box controls (**Destination** field), e-mail addresses (**support@aventail.com**), URLs (**http://www.aventail.com/**), and IP addresses (**165.121.6.26**) are also bold.
Italic	Placeholders that represent information the user must insert.
"To Do" Procedures	Underlined *To Do* headings indicate procedures and step-by-step directions. Multi-step procedures are numbered; single-step procedures are bulleted.

Technical Support

If you experience problems installing, configuring, or running AutoSOCKS refer to any of the following:

- The Aventail web site, **http://www.aventail.com/**, for the latest list of known problems.
- The README.TXT documentation for additional information not contained in the manual.

If necessary, report problems to Aventail using the Bug Report form at the Aventail web site.

Aventail Technical Support:

Web site: **http://www.aventail.com/**

E-mail: **support@aventail.com**

Phone: 206.777.5640

Fax: 206.777.5656

About Aventail Corporation

Aventail Corporation is the leading vendor of next-generation Internet security systems. Its software allows organizations to secure their networks, manage their employees' access to the Internet and build Virtual Private Networks (VPNs). Creating a VPN gives organizations the ability to dynamically create a private communication or data channel over the Internet. Aventail's adherence to open security standards simplifies VPN deployment, enables interoperability, and leverages corporations' existing network investments. Its VPN solutions allow companies to extend the reach of their corporate Intranets to customers, partners, remote offices, and worldwide employees.

Aventail Corporation

117 South Main Street

4th Floor

Seattle, WA 98104-2540

Phone: 206.777.5600

Fax: 206.777.5656

http://www.aventail.com/

info@aventail.com

AutoSOCKS v2.1 *Administration and User's Guide*

This section includes procedural and background information on installing AutoSOCKS to both single and networked workstations. It includes:

- Getting Started with brief explanations of network security and communications
- Definitions of SOCKS and AutoSOCKS
- AutoSOCKS platform and installation requirements
- Installing AutoSOCKS, including network diagrams of Aventail VPN, Aventail Internet Policy Manager, and SOCKS v4-based server configurations
- Creating and editing configuration files

Note: Aventail understands the importance of a flexible, easy-to-use installation process. If you have feedback regarding the AutoSOCKS installation procedures, or if there are additional features you wish to see implemented, please e-mail comments to **support@aventail.com**. Your input is appreciated.

Getting Started

If you're new to AutoSOCKS technology, the following section will help you understand what AutoSOCKS is and does, as well as its relationship to network security in general.

Network Security in a Nutshell

Escalating threats of computer viruses and increased potential for unwelcome hackers are forcing companies to seek ways to safeguard their corporate networks and the information they exchange. The first response to these concerns has been the development of security firewalls—software barriers that control the flow of information. But firewalls can't easily be configured to handle complex security issues such as monitoring network usage, providing private communication over public networks, and enabling remote users to gain secure access to internal network resources.

Enter SOCKS v5, an Internet Engineering Task Force (IETF)-approved security protocol targeted at securely traversing corporate firewalls. It was originally developed in 1990, and is now maintained by NEC. SOCKS acts as a circuit-level proxy mechanism that manages the flow and security of data traffic to and from your local area network or intranet. A workstation whose traffic is proxied by SOCKS is considered "socksified." SOCKS is more than a standard security firewall. It also features:

- Client Authentication: (SOCKS v5 only) Authentication allows network managers to provide selected access to internal and external areas of a network.
- Traffic Encryption: (SOCKS v5 only) Encryption ensures that network traffic is private and secure.
- UDP Support: (SOCKS v5 only) User Datagram Protocol (UDP) has traditionally been difficult to proxy with the exception of SOCKS v5.
- Cross-Platform Support: Unlike most UNIX security solutions, SOCKS code can easily be ported to platforms such as Windows NT, Windows 95, and Macintosh systems.

What is AutoSOCKS?

AutoSOCKS automates the "socksification" of client applications, making it simple for workstations to take advantage of the SOCKS v5 protocol. When you run AutoSOCKS on your system, it automatically routes appropriate network traffic from a WinSock application to the SOCKS server. (WinSock is a Windows component that connects a Windows PC to the Internet using Transmission Control Protocol/Internet Protocol—TCP/IP.) The SOCKS server then sends the traffic to the Internet or the external network. Your network administrator defines sets of rules by which this traffic is to be routed.

AutoSOCKS is designed to run transparently on each workstation. In most cases, you'll interact with AutoSOCKS only when it prompts you to enter authentication information for a connection to a secure SOCKS server. You may also occasionally need to start and exit AutoSOCKS, although network administrators often configure it to run automatically at startup.

To understand AutoSOCKS, you first need to understand a few basics of TCP/IP communications.

TCP/IP Communications

Windows TCP/IP networking applications such as e-mail or ftp use WinSock to gain access to the network or the Internet. WinSock (Windows Sockets) is the core component of TCP/IP under Windows. (TCP/IP is a suite of protocols that the Internet uses to provide for services such as e-mail, ftp, and telnet.)

WinSock Connection to A Remote Host

Via WinSock, an application goes through the following steps to connect to a remote host on the Internet or corporate intranet:

1. The application executes a Domain Name System (DNS) lookup to convert the hostname into an Internet Protocol (IP) address. If the application already knows the IP address, this step is skipped.
2. The application requests a connection to the specified remote host. This causes the underlying stack to begin the TCP handshake. (The TCP handshake is the process by which two computers initiate communication with each other.) When the handshake is

complete, the application is notified that the connection is established, and that data may now be transmitted and received.

3. The application sends and receives data.

What Does AutoSOCKS Do?

AutoSOCKS slips in between the Windows TCP/IP application and the single access point—WinSock. In simple terms, AutoSOCKS redirects WinSock calls (both parameters and data) and reroutes them through a SOCKS-based server when required. The routing is determined by the rules described in the configuration file created when AutoSOCKS is installed. (See "Configuring AutoSOCKS.")

Because AutoSOCKS intercepts calls to WinSock, AutoSOCKS must duplicate WinSock functionality. Since AutoSOCKS also makes calls directly into WinSock, it must behave as a typical WinSock application as well. (See Figure 1.)

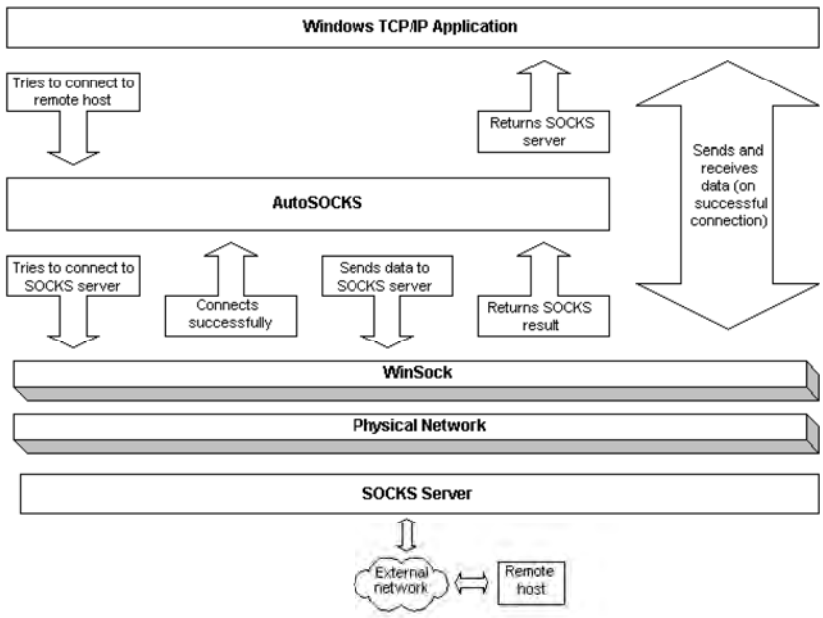


Figure 1. Network application calls intercepted by AutoSOCKS

With AutoSOCKS running, an application executes additional steps in order to connect to a remote host through WinSock. These steps must be transparent to the application so that it cannot differentiate between when AutoSOCKS is running and when it is not. The following three steps are identical to standard WinSock communications steps described above; however, nested inside them are additional actions and options introduced by AutoSOCKS.

1. The application does a DNS lookup to convert the hostname to an IP address. However, if the application already knows the IP address, this entire step is skipped. Otherwise, AutoSOCKS does the following:

   - If the hostname matches a local domain string or does not match a redirection rule, AutoSOCKS passes the name resolution query through to the TCP/IP stack on the local workstation. The TCP/IP stack then performs the lookup as if AutoSOCKS is not running.
   - If the DNS proxy option is disabled, AutoSOCKS allows the request to go through unchanged.
   - If the destination hostname matches a redirection rule domain name (i.e. the host is part of a domain we are proxying traffic to) then AutoSOCKS creates a false DNS entry (HOSTENT) that it can recognize during the connection request. AutoSOCKS will forward the hostname to the SOCKS server in step 2 and the SOCKS server performs the hostname resolution.
   - If the DNS proxy option is enabled and the domain cannot be looked up directly, AutoSOCKS creates a fake DNS entry that it can recognize later, and returns this to the calling application. The false entry tells AutoSOCKS that the DNS lookup should be proxied, and that it should send the fully qualified hostname to the SOCKS server with the SOCKS connection request.

2. The application requests a connection to the remote host. This causes the underlying stack to begin the TCP handshake. When the handshake is complete, the application is notified that the connection is established and that data may now be transmitted and received. AutoSOCKS does the following:

   a. AutoSOCKS checks the connection request.

      - If the request contains a false DNS entry (from step 1) it will be proxied.
      - If the request contains a real IP address and the rules in the configuration file say it should be proxied, AutoSOCKS calls WinSock to begin the TCP handshake with the server designated in the config file.
      - If the request contains a real IP address and the configuration file rules says that it should *not* be proxied, the request is passed to WinSock and processing jumps to step 3 as if AutoSOCKS is not running.

   b. When the connection is completed, AutoSOCKS begins the SOCKS negotiation.

      - It sends the list of authentication methods enabled in the configuration file.
      - Once the server chooses an authentication method, AutoSOCKS executes the specified authentication processing.
      - It then sends the proxy request to the SOCKS server. This includes either the IP address provided by the application or the DNS entry (hostname) provided in step 1.

c. When the SOCKS negotiation is completed, AutoSOCKS notifies the application. From the application's point of view, the entire SOCKS negotiation including the authentication negotiation, is merely the TCP handshaking.

3. The application transmits and receives data.

If an encryption module is enabled and selected by the SOCKS server, AutoSOCKS encrypts the data on its way to the server on behalf of the application. If data is being returned, AutoSOCKS decrypts it so that the application sees clear text data.

AutoSOCKS Platform Requirements

AutoSOCKS runs under Windows 3.1, Windows for Workgroups 3.11, Windows 95, and Windows NT 3.51 and 4.0. These five platforms can be divided into two groups. Operating requirements and interface features unique to each group are described below.

Windows 95 and Windows NT 4.0

Windows 95 and Windows NT 4.0 have virtually identical interfaces. AutoSOCKS commands are accessed in the Programs list located on the Start menu and from the minimized AutoSOCKS icon on Taskbar tray.

System Requirements

AutoSOCKS system requirements for Windows 95 and Windows NT 4.0 include the following:

- Pentium-based personal computer
- Windows 95 or Windows NT 4.0
- 16 MB application RAM (8 MB on Windows 95)
- 3.5 MB hard disk space
- 16- or 32-bit WinSock-based TCP/IP application(s)
- Network-accessible SOCKS v4 or SOCKS v5 compliant server
- A WinSock compatible TCP/IP stack needs to be installed and configured prior to running AutoSOCKS. This can be the Microsoft-provided TCP/IP stack or a third-party TCP/IP stack.

Interface Features

- The AutoSOCKS program icon can be accessed via the Start menu, Programs option, and Aventail AutoSOCKS menu command.
- When AutoSOCKS is running in the background, the AutoSOCKS icon is visible in the system tray (unless the Hide Icon command is enabled).
- The AutoSOCKS system menu can be displayed by right-clicking the AutoSOCKS icon located in the Taskbar tray.
- AutoSOCKS can be uninstalled via the Start menu by using the **Add/Remove Programs** icon in the Control Panel folder.

Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51

Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51 have similar interfaces. AutoSOCKS commands are accessible from the Aventail AutoSOCKS program group and from the minimized icon's System menu when AutoSOCKS is running.

System Requirements

AutoSOCKS system requirements for Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51 include the following:

- 486-based personal computer
- 4 MB application RAM for Windows 3.1 and Windows for Workgroups 3.11; 16 MB for Windows NT
- 3.5 MB hard disk space
- 16- or 32-bit WinSock-based TCP/IP application(s)
- Network-accessible SOCKS v4 or SOCKS v5 compliant server
- A WinSock compatible TCP/IP stack needs to be installed and configured prior to running AutoSOCKS. This can be the Microsoft-provided TCP/IP stack or a third-party TCP/IP stack.

Interface Features

- The AutoSOCKS program icon is accessed via the AutoSOCKS program group window in Program Manager.
- The AutoSOCKS system menu is displayed by clicking the AutoSOCKS icon located in the AutoSOCKS program group.
- AutoSOCKS can be uninstalled using the Uninstall icon in the AutoSOCKS program group window.
- When AutoSOCKS is running in the background, the AutoSOCKS icon is minimized on the desktop (unless the Hide Icon command is enabled)

Installation Source Media

Regardless of platform, AutoSOCKS can be delivered on CD; in a network-delivered, self-extracting archive file; or on diskette.

This runs SETUP.EXE and installs AutoSOCKS. You can specify an installation directory, or AutoSOCKS will install in the default AutoSOCKS directory.

- **CD**: The CD contains the AutoSOCKS installation program, SETUP.EXE. It also contains in the \DOCS directory the AutoSOCKS v2.1 *Administration and User's Guide* formatted for Acrobat Reader.

- **Network Delivered Source Media**: The network-delivered source media is a self-extracting archive containing the required disk/directory structure within the archive file. The archive filename will be similar to AS21ED.EXE.
- **Diskette Based Source Media.** The diskette based source media is composed of two separate disks (labeled Disk 1 and Disk 2) that contain all of the AutoSOCKS installation files.

Installing AutoSOCKS

AutoSOCKS can be installed to a single workstation or to multiple networked workstations. In either case, you must perform an initial installation of the software and create one or more configuration files. This procedure is described under "Individual Installation." Once the initial installation is complete, you can then install to a series of networked computers using the instructions and information described under "Network Installation."

Note: Check the Quick Start Card for an easy-to-follow guide to individual workstation installation.

Configuration Files

Integral to the initial installation of AutoSOCKS is deciding how SOCKS traffic should be redirected through the network. Network redirection rules (used to determine if and how SOCKS redirection should occur) are defined in the AutoSOCKS configuration file. Configuration files are initially created at the end of the installation process; however; they can be added, edited, and removed at any time using the Config Tool (in Windows 3.1, Windows for Workgroups 3.11, or Windows NT 3.51 via the System menu in the Aventail Program Group; in Windows 95 or Windows NT 4.0 via the Aventail icon in the Taskbar tray). The process of creating one or more configuration files is described under "Configuring AutoSOCKS."

If you are installing AutoSOCKS on multiple networked workstations, refer to "Network Installation" to determine the best method for maintaining and distributing configuration files. You can then proceed through the initial installation. An Installation Wizard will guide you through the steps, culminating with the option to create a configuration file.

Individual Installation

To install AutoSOCKS

Before running Setup, it is advisable to close all open Windows applications.

1. Installation procedures vary slightly, depending on which media source you use:
   - If you are installing directly from CD-ROM, run SETUP.EXE from the AutoSOCKS directory (\AS_v21).
   - If you are installing directly from diskette, run SETUP.EXE on disk 1.

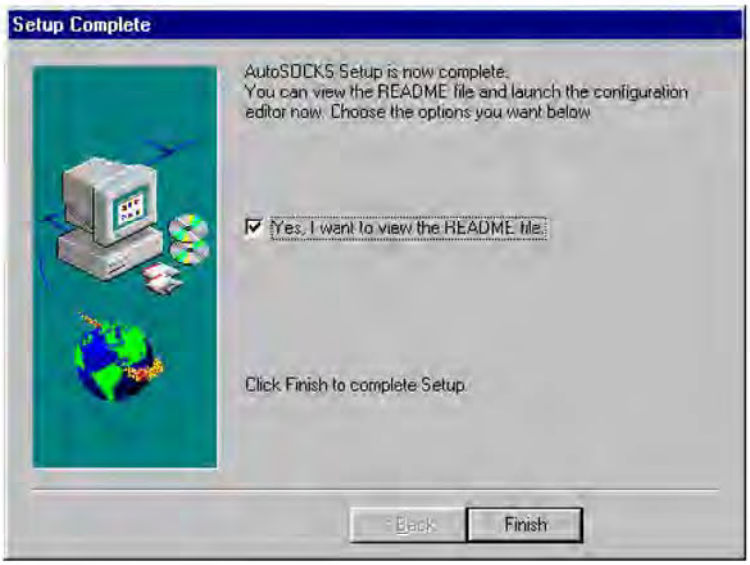
- If you are installing from a network-delivered self-extracting archive, simply run the archive file. This will extract the installation files and automatically launch the setup program.

The AutoSOCKS Installation Wizard then guides you through the process of installing the AutoSOCKS application.

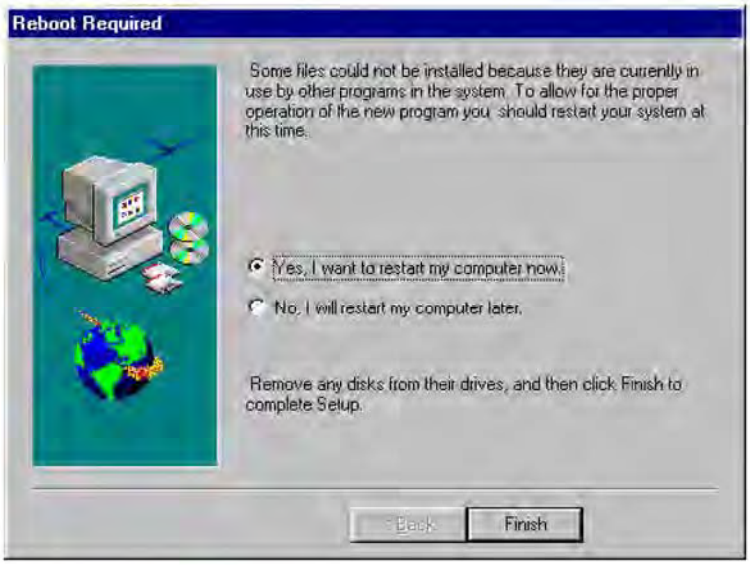
2. At the end of the Setup Program you can click the **Yes, I want to view the README file** box in the Setup Complete dialog box. This opens the README file for the latest information on AutoSOCKS.

   -OR-

   Simply click the **Finish** button to complete the Setup Program.



3. The setup program will then ask you if you want to restart now or later.



4. After restarting your PC, start AutoSOCKS for the first time.
5. AutoSOCKS will ask you if you want to run the Configuration Wizard.

   If you select **Yes,** then the Configuration Wizard will launch to help you create a new configuration file.

   If you select **No**, then AutoSOCKS will ask you to select a configuration file to use.
6. After creating or selecting a configuration file, AutoSOCKS will now be finished installing.

To uninstall AutoSOCKS

The procedure to uninstall (remove) AutoSOCKS varies depending on whether you are running a 16- or 32-bit Windows operating system.

- To uninstall AutoSOCKS from Windows 95 and Windows NT 4.0, double-click **Add/Remove Programs** in the Control Panel window, select AutoSOCKS from the list of programs on the Install/Uninstall tab, and click the **Add/Remove** button.
- To uninstall AutoSOCKS on Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51, use the Uninstall icon in the AutoSOCKS program group.

Network Installation

In general, the process of installing AutoSOCKS to multiple networked workstations involves selection of a file server to use, creation of a staging area for the AutoSOCKS software, and copying the AutoSOCKS files to a shared network directory from the source media. Additional

options include adding a default configuration file, and creating a universal batch/script file that specifies required default command line options when executed by the end user or installation personnel. AutoSOCKS files should be placed in a network drive which can be accessed as a mapped drive or, for Microsoft networks, via a UNC path name (**\\computer_name\share_name\AutoSOCKS**).

Networked Configuration File Setup

There are a number of ways to set up networked client configuration files. These are the most common:

- Client configuration file distributed via a mapped network drive (Novell or Microsoft)
- Client configuration file distributed via a Microsoft UNC path and filename
- Local client configuration file common for all users, but distributed via the standard AutoSOCKS installation and upgrade program

Administrator-Maintained Shared Configuration Files

This is the most desirable configuration method—multiple workstations sharing one or more administrator-maintained configuration files located in a common directory. It is an easily managed configuration because the configuration file is maintained by the network administrator and changes to network topology can be reflected quickly via network distribution. For example:

- A single-networked (usually read-only) configuration file is shared by more than one client workstation. This method is appropriate when workstations share identical message traffic routing rules.
- Multiple configuration files are shared by multiple workstations. This option is useful when you have workstations organized into functional groups (engineering, marketing, accounting, etc.) with group-specific message traffic routing rules.

Shared Configuration File Distribution

Shared configuration files can be easily distributed and, if necessary, updated via the network. All configuration files should be tested first before being distributed.

To distribute a shared configuration file

There are three methods for distributing shared configuration files.

- Copy the file to a Microsoft or Novell network mapped drive accessible by all users. Make sure that end users configure their AutoSOCKS clients to load the configuration file located on the mapped drive.

-OR-

- Copy the file to a Microsoft Windows workstation supporting UNC-sharing for file resources. (Both the 16- and 32-bit AutoSOCKS clients support specification of the configuration file using the Microsoft UNCs.)

This distribution method has all the benefits of placing the file on a network mapped drive with the added bonus of convenience—end users don't have to actually map the network drive.

-OR-

- Create a shared configuration file, AUTOSOCK.CFG, to be installed on workstations during the standard AutoSOCKS installation/upgrade process. (Place the shared configuration file into the DISK1 directory.) Whenever the AutoSOCKS client is installed or updated, it will to automatically copy AUTOSOCK.CFG to the end user's workstation and set AutoSOCKS to use it.

  Note: If a configuration file is specified as a command line option in the Setup program, installation of the AUTOSOCK.CFG configuration file will be overridden.

Setup Command Line Options

The AutoSOCKS setup program accepts several command line options which allow you to customize the installation process. By using options on the command line, installation can either run entirely unattended, or it can be used to specify a network-based AutoSOCKS configuration file. Each of the command line options are listed in the following table along with a brief explanation. Specifying any of the options that support unattended mode will cause the setup program to perform an automatic installation using default values for any options not explicitly specified.

Option	Explanation	Default Value	Unattended
config=*path*	Specifies the location of the AutoSOCKS configuration file. The destination can be either a local file, or can be specified with a UNC filename or common mapped drive.	Nothing	No
dir=*path*	Specifies the directory containing AutoSOCKS installation files.	**C:\Program Files\Aventail\ AutoSOCKS**	Yes
autostart	If specified, moves the AutoSOCKS application into the Startup group; otherwise AutoSOCKS must be started manually.	Don't put in startup	Yes
nocfg	Specifies that none of the configuration tools should be installed. This option will keep the Config Tool and Configuration Wizard from being installed.	Configuration tools are installed	No
nt=16|32|both	Selects the type of WinSock applications supported by AutoSOCKS: 16-bit, 32-bit or both. This option is only valid for Windows NT	Both	Yes

Configuring AutoSOCKS

Configuration files are created using the Config Tool application. This application can be launched during AutoSOCKS installation or any time you wish to add, modify, or remove a configuration file.

The steps for creating a new configuration file are:

1. Define the SOCKS servers
2. Define the destinations (networks and hosts)
3. Specify redirection rules
4. Enter Local Name Resolution (optional)
5. Manage authentication modules

These procedures are described in the text below.

To launch the Config Tool

The Config Tool opens with the Open AutoSOCKS Configuration File dialog box. After a configuration file is selected or a new file name is entered, the main window of the Config Tool appears.

1. Click the **Yes, I want to configure AutoSOCKS** box in the Setup Complete dialog box (during installation).

   -OR-

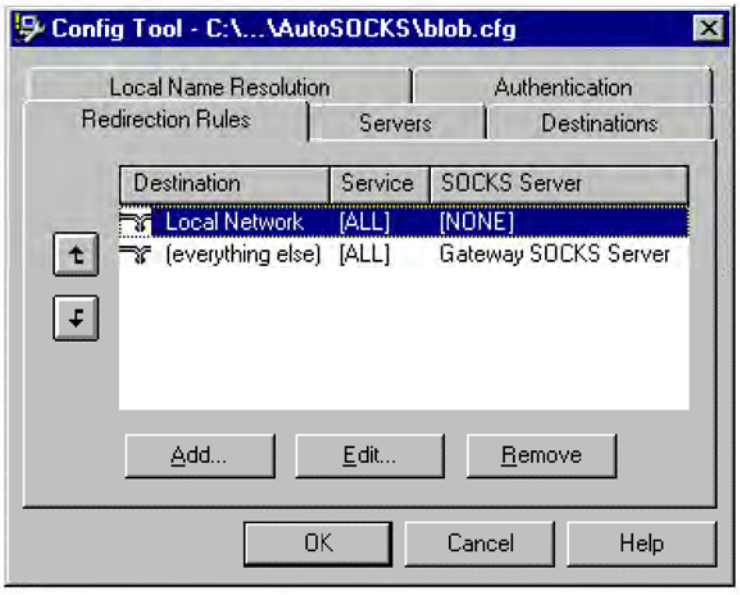
   Select Config Tool from the Aventail AutoSOCKS program group (Windows 3.1, Windows for Workgroups 3.11, or Windows NT 3.51) or the Aventail AutoSOCKS menu (Windows 95 or Windows NT 4.0 Programs menu option).

2. If you are creating a new configuration file, enter a name for the configuration file. (AutoSOCKS defaults to AUTOSOCK.CFG).

   -OR-

   Select the configuration file you wish to open.

   This displays the main window of the Config Tool.



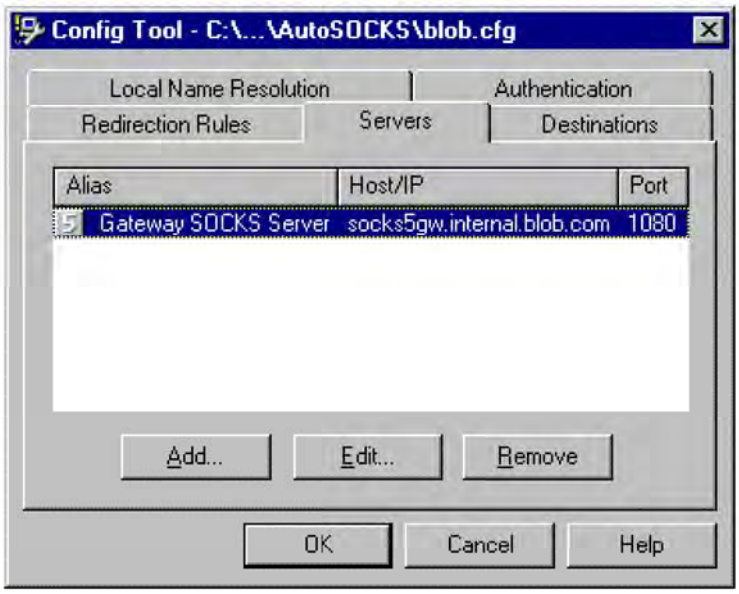
The Config Tool window contains five tabs. The properties defined on each tab can be edited at any time.

Tab	Function
Redirection Rules	Specifies how network requests are routed to the SOCKS servers.
Servers	Defines the SOCKS servers.
Destinations	Specifies the network and host addresses that should be routed through SOCKS servers.
Local Name Resolution	(Optional) Specifies hostnames that will be resolved by the local workstation.
Authentication	Enables, disables, and sets properties for the authentication modules.

You can change the width of any of the fields on the tabs by moving the cursor to the dividing line between the fields on the field bar. When the cursor changes to a double-headed arrow, click and drag to resize the field.

Define a SOCKS Server

SOCKS servers are defined on the Servers tab in the Config Tool.

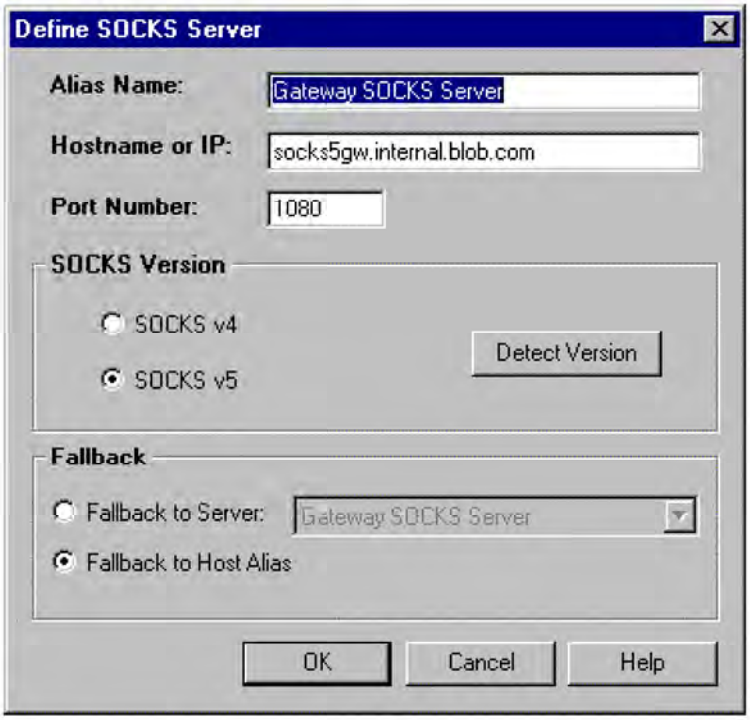


Field	Definition
Alias	The descriptive name you assign to the server. (The number is the SOCKS version.)
Host/IP	The hostname and/or IP address of the server.
Port	The port on which the server is listening.

To add a SOCKS server

1. On the Server tab, click the **Add** button.

   The Define SOCKS Server dialog box appears.



Field	Definition	
Alias Name	User-friendly alias for SOCKS server.	
Hostname or IP	Actual hostname or full numeric IP address for SOCKS server.	
Port Number	SOCKS server port. Default value is 1080.	
SOCKS Version	SOCKS v4:	SOCKS Version 4.0
	SOCKS v5:	SOCKS Version 5.0
	Detect Version:	Detect SOCKS version number.
Fallback	Fallback to Server:	SOCKS server alias for redundant server
	Fallback to Host Alias:	Use DNS records for redundancy

2. In the Alias Name box, type a user-friendly alias for the SOCKS server.
3. In the Hostname or IP box, type the actual hostname of the SOCKS server or its IP address.
4. In the Port Number box, type the SOCKS server's port number. If you don't enter a value, it defaults to the standard SOCKS port 1080.
5. Under SOCKS Version, select the version of SOCKS supported by the server. If you're unsure of the version, click the **Detect** button.

   Note: Typically you should select SOCKS v5 unless the server can only support SOCKS v4.
6. Under Fallback, directly specify a SOCKS server for redundancy or use the Host Alias to specify a SOCKS server.

To edit SOCKS server properties

- Select the SOCKS server you want to edit and click the **Edit** button.

  The Define SOCKS Server dialog box appears with the selected server data filled in. Edit any of the information.

To remove a SOCKS server definition

- Select the SOCKS server you want to remove and click the **Remove** button.

  The server is deleted from the list. Corresponding redirection rules will also be deleted.

Define a Destination

Destinations are defined on the Destinations tab in the Config Tool.

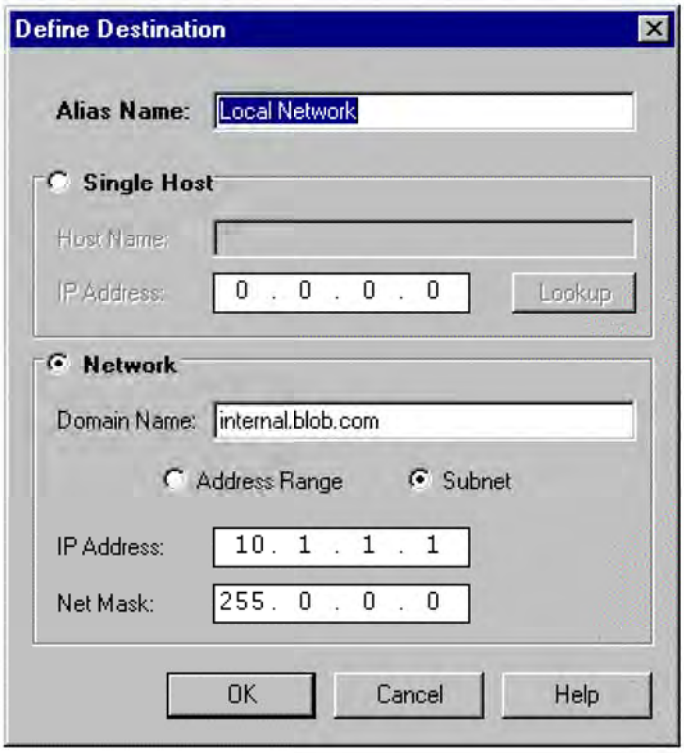
After one or more SOCKS servers are defined, destinations to be routed through them should be added.

Note: The **(everything else)** destination refers to all network and host addresses not otherwise defined.

To add a destination

1. On the Destinations tab, click the **Add** button.

   The Define Destination dialog box appears.



Field	Definition	
Alias Name	User-friendly alias for destination network or host	
Single Host	A specific destination computer	
	Hostname:	Actual name of destination network or host
	IP Address:	Full numeric IP address
	Lookup:	Look up IP address
Network	One or more computers in a network	
	Domain Name:	Domain of the network
	Address Range:	Beginning and ending IP addresses
	Subnet:	IP address and netmask
	From:	Address Range: Starting IP address. Subnet: IP address
	To:	Address Range: Ending IP address. Subnet: Net mask

2. In the Alias Name box, type a user-friendly alias to use for the destination network or host.
3. Choose either the Single Host or Network option:

   Under Single host, type the actual name of the host system and/or its full, numeric IP address. If you don't know the Host's IP address, the **Lookup** button will help you locate it.

   -OR-

   Under Network, type the domain of the network and choose either the Address Range or Subnet options:

Use	To
Address Range	Enter a starting and ending IP address. All addresses between the two will be included as part of the destination. For example, a starting IP address of 192.168.1.0 and an ending IP address of 192.168.1.255 would include all hosts on the 192.168.1 subnet.
Subnet	Enter an IP address and a net mask. This is another way to specify a group of destinations. For example, an IP address of 192.168.1.0 and a net mask of 255.255.255.0 defines the same address range as shown above.

To edit a destination

- Select the destination you want to edit and click the **Edit** button.

  The Define Destination dialog box appears with the selected destination data filled in. Edit the data as necessary.

To remove a destination

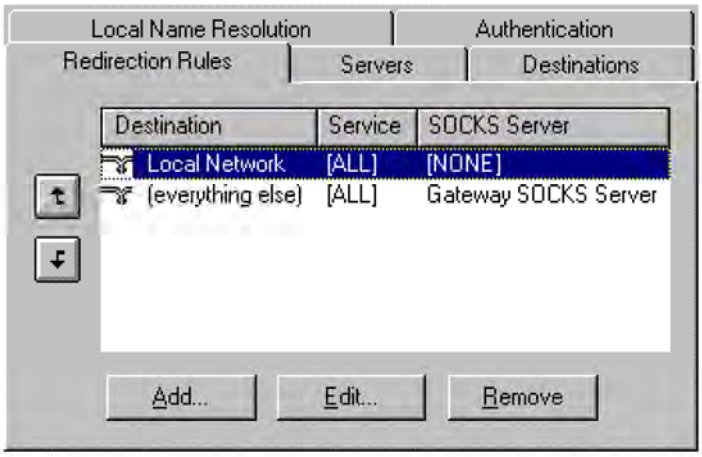
- Select the destination you want to remove and click the **Remove** button.

  The destination is deleted from the list. The corresponding redirection rule will also be deleted.

Enter Redirection Rules

Once servers and destinations are defined, you can then specify how you want AutoSOCKS to redirect (or deny) access to various hosts and services such as e-mail, FTP, and HTTP.

Redirection rules are specified on the Redirection Rules tab in the Config Tool.



In the above example, the redirection rules specify that network traffic on the Local Network will not be redirected through a SOCKS server. All traffic not directed to the Local Network will be proxied through the Gateway SOCKS Server.

Field	Definition
Destination	Destinations defined on the Destination tab
Service	Type of Internet traffic
SOCKS Server	Servers defined on the Server tab

You can change the width of any of the three fields by moving the cursor to the dividing line between the fields on the field bar. When the cursor changes to a double-headed arrow, click and drag to resize the field.

To add a redirection rule

As you add destinations, use the arrow buttons to prioritize them. List the most specific rules first and the general rules last.

Note: AutoSOCKS scans the list from the top down and uses the first matching rule it finds, so it is important to list the most specific rules first.

1. On the Redirection Rules tab, click the **Add** button.

The Define Redirection Rule dialog box appears.

Field	Definition	
Destination	Host or server destination for message traffic.	
Service	Type of Internet traffic.	
	Name or Port No.:	Select from a list of common service ports or enter a new port.
	Use all ports:	Apply the rule to all services.
	TCP and UDP:	Apply the defined rule to both TCP and UDP traffic.
	TCP only:	Apply the defined rule to TCP traffic only.
	UDP only:	Apply the defined rule to UDP traffic only.
Proxy Redirection	Specify how to redirect traffic.	
	Redirect via:	Redirect all traffic through the SOCKS server selected from the list.
	Do not redirect:	Route traffic directly to the specified destination without being redirected through SOCKS.
	Deny service:	Deny access to the specified destination. The network connection is blocked locally instead of at the server level.

2. Select a destination from the Destination list.
3. Under Service, check the **Use all ports** box to apply the rule to all services. Otherwise, select an individual service from the **Name or Port No.** list.
4. Under Proxy Redirection, select one of three redirection options:

Note: If you select Deny Service and the user has edit control of the Config file, the option can be circumvented by quitting AutoSOCKS or by changing the option in the dialog box.

To edit a redirection rule

- Select the redirection rule you want to edit and click the **Edit** button.

The Define Redirection Rule dialog box appears with the selected data filled in. Edit any of the information.

To remove a redirection rule

- Select the redirection rule you want to remove and click the **Remove** button.

The redirection rule is deleted from the dialog box.

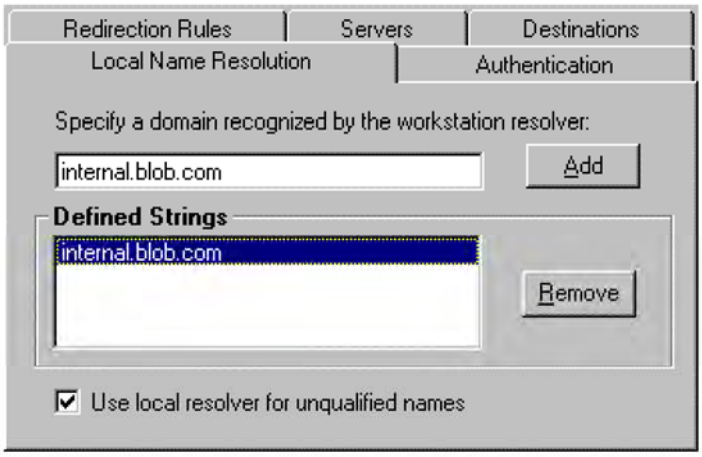
Define Local Name Resolution

Local Name Resolution instructs AutoSOCKS to resolve hostnames locally without needing to venture on to the Internet. This optional feature offers you another level of control over how AutoSOCKS performs name resolution.

The local workstation resolver is the name resolution component of the local TCP/IP stack. This feature acts as a shortcut; hostnames matching the strings defined in the Local Name Resolution dialog box are passed to the local resolver for name resolution instead of being proxied through the SOCKS v5 server.

For example, if **internal.blob.com** is added to the Defined Strings list, then a workstation attempting to connect to **www.internal.blob.com** would perform hostname resolution using the local TCP/IP stack.

Local Name Resolution is specified on the Local Name Resolution tab in the Config Tool.



Field	Definition
Specify Domain	New domain name
Defined Strings	List of domain names that can be resolved locally
Use local resolver	Pass through unqualified hostnames to the local resolver

To add a local domain name

- On the Local Name Resolution tab, type the new name in the Specify Domain text box and click the **Add** button.

  The new name is moved into the Defined Strings text box. It is now active.

To remove a local name

- Select the domain name you want to remove from the Defined Strings text box and click the **Remove** button.

  The domain name is removed from the list.

Managing Authentication Modules

SOCKS v5 servers often require user authentication before allowing access. AutoSOCKS authentication modules facilitate this process by displaying dialog boxes which ask for username and password information as well as other authentication credentials.

The current AutoSOCKS authentication modules (SOCKS v4 Identification, Username/Password, Challenge Handshake Authentication Protocol, and Secure Socket Layer) support an AutoSOCKS feature known as credential caching. Credential caching is the process of retaining your authentication credentials once they've been accepted by the SOCKS server. Using credential caching, you can enter your credentials for a SOCKS server once per AutoSOCKS session, rather than once for each individual connection (a tedious task for applications such as WWW browsers).

AutoSOCKS can cache authentication credentials in memory, based on the option you select in the Authentication dialog box. Memory caching stores the credentials for the current session only. When you restart AutoSOCKS or Windows, the memory cache is flushed and you must renter your credentials as prompted.

Authentication modules are managed and configured on the Authentication tab in the Config Tool.

Field	Definition
Module Name	The name of the authentication module on disk;. <Null Auth> indicates that no authentication module will be used.
Description	The description of the authentication method.
Indicator	Check this option to display a visual indication of the authentication/encryption being used as network traffic is generated.

Each authentication module includes its own module-specific configuration. To view or edit a module's configuration dialog box, select the module from list on the Authentication tab and then click the **Setup** button.

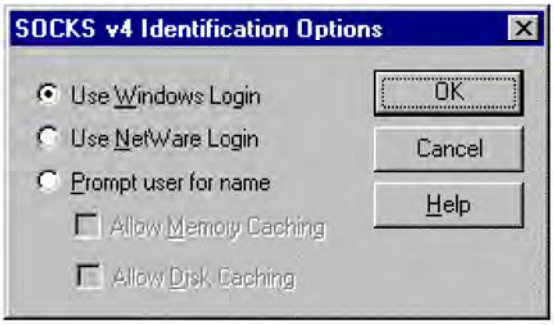
Authentication modules can be selectively enabled and disabled using the Disable/Enable button. By default, the modules are all enabled. This is indicated by the green button next to the module name. When a module is disabled, the button is red.

To configure the SOCKS v4 Identification module

AutoSOCKS includes backward compatibility for the SOCKS v4 protocol. SOCKS v4 does not support password authentication; only your username is sent unencrypted to the SOCKS server along with your connection request. Your username is determined by entries in the SOCKS v4 Identification Module configuration dialog box.

1. On the Authentication tab in the Config Tool, select **sv4auth** (SOCKS v4 Authentication) and click the **Setup** button.

   The SOCKS v4 Identification dialog box appears.



Field	Description	
Use Windows Login	Identify users by their Windows Login names.	
Use NetWare Login	Identify users by their Novell NetWare login names.	
Prompt user for name	Identify users by the names they enter for this specific purpose.	
	Allow Memory Caching	Stores credentials in memory for this session only. Cache is flushed upon restart; credentials must be reentered as prompted.
	Allow Disk Caching	This option is currently unavailable. (Stores credentials on disk for future sessions.)

2. When the option **Prompt user for name** is selected, choose the desired caching option. (Currently only Memory Caching is available.)
3. After making appropriate selections, click **OK**.

   The dialog box closes and the Config Tool is displayed.

To configure the Username/Password authentication module

AutoSOCKS supports the RFC 1928 (Internet standards document) username and password authentication protocol. This authentication method sends your username and password *in clear text* across the network to the destination server. The Username/Password authentication module dialog box contains only credential caching options.

1. On the Authentication tab in the Config Tool, select **unpw** (Clear text username/password) and click the **Setup** button.

   The Username/Password dialog box appears.

Field	Description
Allow Memory Caching	Stores credentials in memory for this session only. Cache is flushed upon restart, credentials must be reentered as prompted.
Allow Disk Caching	This option is currently unavailable. (Stores encrypted credentials on disk for future sessions.)

2. The selection defaults to **Allow Memory Caching**. Click **OK**.

   The dialog box closes and the Config Tool is displayed.

To configure the CHAP Authentication module

AutoSOCKS supports the Challenge Handshake Authentication Protocol (CHAP). This authentication method sends your username and password *encrypted* across the network to the destination server. The CHAP authentication module dialog box contains only credential caching options.

1. On the Authentication tab in the Config Tool, select **chap** (CHAP) and click the **Setup** button.

   The CHAP Options dialog box appears.

Field	Description
Allow Memory Caching	Stores credentials in memory for this session only. Cache is flushed upon restart, credentials must be reentered as prompted.
Allow Disk Caching	Currently Unavailable. (Stores encrypted credentials on disk for future sessions.)

2. The selection defaults to **Allow Memory Caching**. Click **OK**

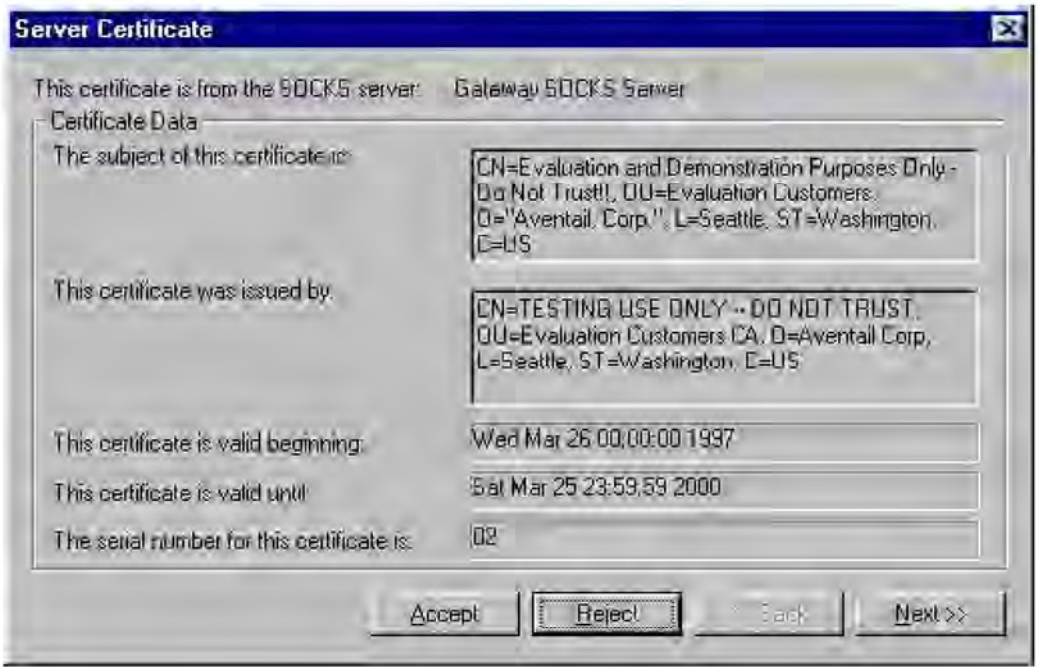
The dialog box closes and the Config Tool is displayed.

To configure the SSL security module

AutoSOCKS supports Secure Socket Layer (SSL) v3.0, a session-layer protocol for securing connections in a general, protocol-independent fashion. At this time, SSL is a TCP-only enhancement; when using SSL with UDP associations, the bulk data is passed without protection.

Normally SSL servers are required to have an RSA key pair and a certificate. RSA is a public/private-key cryptographic system; it creates a key pair: a private key (which, as the name suggests, is kept absolutely private and never shared) and a public key (which is widely published.)

However, you normally must then establish some kind of relationship between your RSA public key and the identity of the server, so that somebody else cannot create their own RSA key information and use it to impersonate your server. *Certificates* establish this relationship. A certificate is essentially an electronic "statement" which verifies that a certain RSA public key is associated with a particular name.



Certificates are issued by a Certification Authority (CA), and are linked together to form a construct called a certificate *chain of authorities*, each one having a previous entity vouching for its identity. In practice, chains generally include two certificates: one confirming the identity of the server, and the other—a "root" certificate—containing the identity and public key of the CA.

Certificates contain special integrity checks and electronic signatures which verify that the certificate is genuine, was issued by some certification authority, and was not tampered with. Anybody can issue a certificate that says anything; the client must know who issued the certificate, and have some trust relationship in order to believe that it is in fact true. The client has a list of trusted CAs. A set of certificate chains can be structured as a tree, with new certificates stemming from old ones. A base CA is sometimes called the “root” or "trusted root" of this tree.

The SSL module dialog box contains an initial set of options regarding the viewing of certificates. It expands into more detail when the **Advanced** button is clicked.

1. On the Authentication tab in the Config Tool, select **sslclnt** (SSL Security) and click the **Setup** button.

   The SSL Options dialog box appears.

Field		Description
Upon Successful Connection:		The certificate is valid.
	View when the server certificate is new.	Upon successful connection, display the server certificate if it hasn't been displayed during the current session.
	Don't show me the certificate.	Never display the server's certificate if it is deemed valid.
If a server certificate is suspect:		The certificate may not be valid.
	Always show me suspect certificates.	Each time a certificate is deemed suspect by AutoSOCKS, display it.
	Show me the same suspect certificate once.	Once a suspect certificate has been accepted by the user, don't display it again.
	Show me the certificate, but reject the connection.	Reject the connection, but display the suspect certificate.

2. Select an action that AutoSOCKS should take once it deems the server certificate acceptable. (Under normal circumstances, the server will provide AutoSOCKS with a certificate to match with one of AutoSOCKS' trusted roots, if any exist):

   - **View when the server certificate is new:** AutoSOCKS displays the certificate the first time it's seen. Subsequent connections to the same SOCKS server will not cause the certificate to be redisplayed.
   - **Don't show me the server certificate:** AutoSOCKS will never display a valid certificate.

3. Select an action that AutoSOCKS should take if it receives a server certificate that is suspect:

   - **Always show me suspect certificates**: AutoSOCKS will display suspect certificates each time they are received. The certificate dialog box will appear for each new connection to the server(s) sending a suspect certificate. (This option allows you to continue the connection despite the fact that the certificate is questionable.) The SSL module authenticates the server's certificate based on the following questions:

Is the certificate valid?

Did a trusted certificate authority (CA) issue the certificate?

Is the name established by the certificate the same as the name of the server for this connection?

If a certificate does not pass all three tests, it is considered a suspect certificate.

- **Show me the same certificate once:** AutoSOCKS will display a suspect certificate the first time that it is received. If you choose to maintain the connection, the questionable certificate will not be displayed again during the current session.
- **Show me the certificate, but reject the connection:** AutoSOCKS will reject a connection if the certificate is suspect. It will display the certificate to allow you to view it.

4. Clicking the **Advanced** button in the dialog box to expand the dialog box into acceptable cipher (a cryptographic algorithm used to encrypt the data stream) options.

Field	Description	
Allow RC4	Offer the RC4 cipher to the server.	
Allow DES	Offer the DES cipher to the server.	
Allow NULL Encryption	Do no encryption. SSL will be used to authenticate, not encrypt.	
Allow Diffie-Hellman Anonymous	Don't authenticate the server; only do encryption.	
Trusted roots	Choose a file with a certificate that specifies certificate chain roots that are to be trusted.	
	Add	Add a new trusted root.
	Import	Import a trusted root.
	Delete	Delete a trusted root.
	View	View a trusted root certificate file.

During the initial SSL connection negotiation, the client and the server negotiate which cipher to use. Checking a particular cipher in the dialog box doesn't mean that it will be used. Instead, each checked cipher is *offered* to the server, but the server must make the final determination. If the server requires a cipher that isn't selected in this dialog box, the authentication will fail.

Any or all of the acceptable cipher options can be selected:

- **Allow RC4:** AutoSOCKS encrypts the information using the RC4 cipher.
- **Allow DES:** AutoSOCKS encrypts the information using the DES cipher.
- **Allow Null Encryption:** AutoSOCKS allows the server to choose *no* encryption. Message integrity is still assured, but the data will be sent in the clear.
- **Allow Diffie-Hellman Anonymous:** AutoSOCKS will be able to communicate with the SOCKS server without requiring a server certificate. The client and server will not exchange certificates, so there will be no authentication. The encryption will still be negotiated, and the data stream will still be encrypted (unless NULL encryption is chosen by the server).

5. If necessary, add a trusted root to the list of trusted roots by pressing the **Add** button, and selecting a file that contains a trusted root certificate.

   When AutoSOCKS receives a certificate from a server, it looks at the root of the certificate chain and matches it against AutoSOCKS' list of trusted root certificates.

6. After making appropriate selections, click OK.

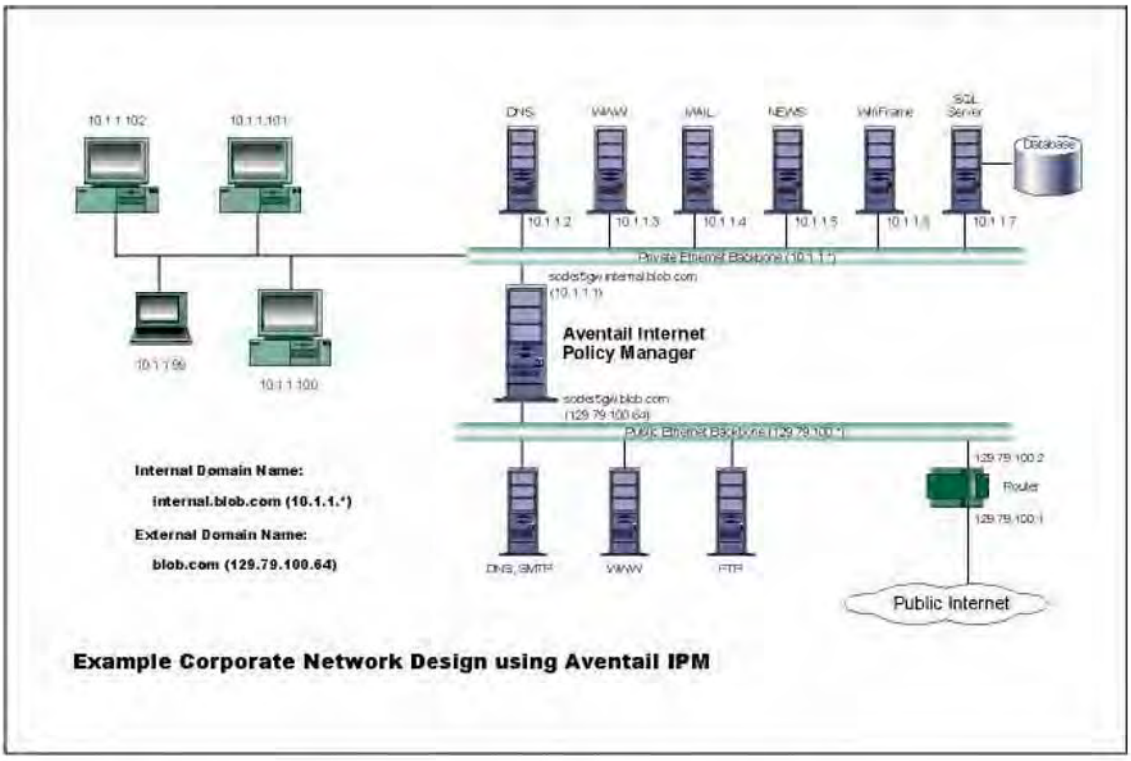
   The dialog box closes and the Config Tool is displayed.

Example Network Configurations

The following sections describe the setup of AutoSOCKS in an example network configuration using the Aventail Internet Policy Manager (IPM) and the Aventail VPN Server.

Configuration Using Aventail Internet Policy Manager

To better describe how to get started configuring AutoSOCKS for use with the Internet Policy Manager, we have created an example network configuration that will be used in all examples throughout this section. Below is an example network topology architecture that emphasizes simplicity to facilitate easy adaptation to real world network designs.



AutoSOCKS in an Aventail Internet Policy Manager Environment

The design used in the example above consists of two individual Ethernet segments, one public and one private. The public segment is used to host anonymous services available to the general public. The public access is provided through a router that is connected to the public Internet. The private segment is used to house all of the corporation's private network resources and data to be used only by internal company employees. To provide protection of the private LAN from unwanted external access, the Aventail IPM is configured such that operating system routing is disabled. Therefore, no direct network connections between the public LAN and the private LAN can be created without being proxied through the Aventail IPM.

The end user workstations (10.1.1.99 through 10.1.1.102) illustrate client workstations, onto which, AutoSOCKS will be deployed. Due to the routing restrictions described above, these clients will have no network access beyond the Aventail IPM unless they are running AutoSOCKS, which will automatically proxy their application traffic. In this situation, AutoSOCKS will forward traffic destined for the Internet to the Aventail IPM. Then, based on the administrative configuration, the Aventail IPM will proxy end user traffic out beyond the boundary on which the Aventail IPM is located. The client workstations used in this example are Microsoft Windows based PC's.

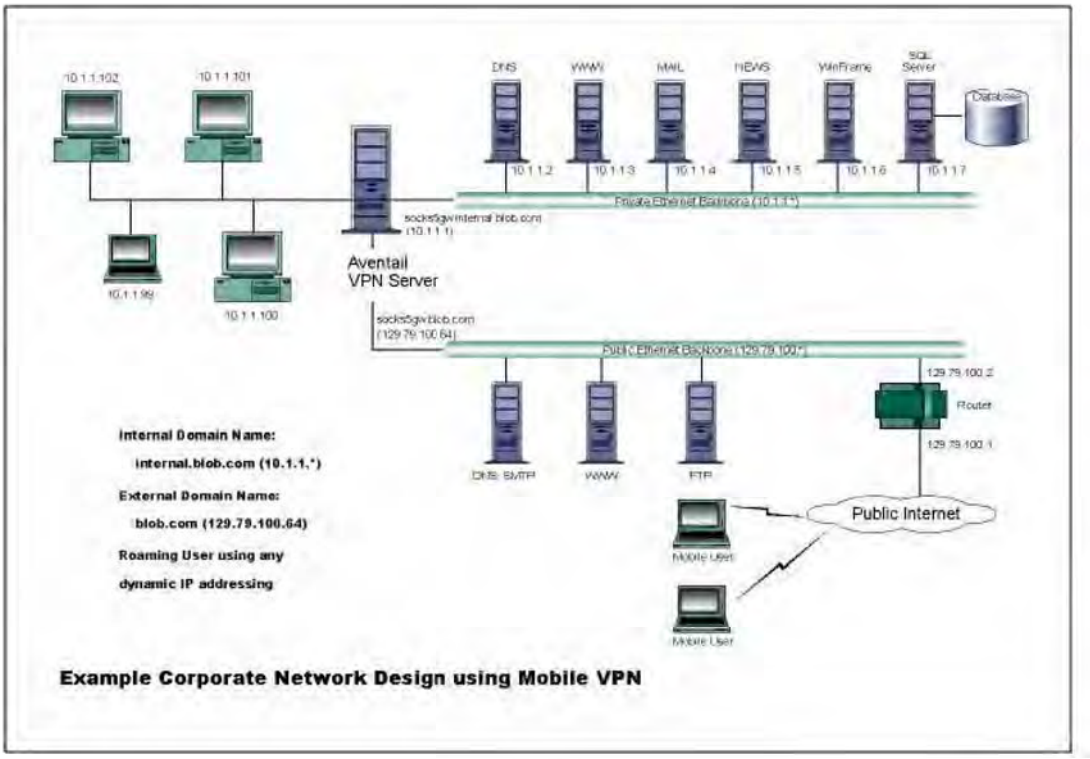
The other servers on the private segment are "internal" or private servers that contain information and tools that are not intended for public use or consumption. If these individual hosts require access beyond the Aventail IPM they can also be configured to use AutoSOCKS. As in the client workstation case, AutoSOCKS will allow applications running on these hosts to traverse the Aventail IPM public/private boundary. In most situations, for more stringent security, these hosts don't have access to the public network at all.

The Aventail IPM in our example, has two network adapters configured to use the internal IP address of 10.1.1.1 and an external address of 129.79.100.64. Since the internal network address space is part of the IANA reserved address space (per BCP RFC 1918) routing MUST be disabled on this host and routing advertisements for this internal network MUST NOT be propagated to the outside world. End user authentication has been enabled on the Aventail IPM server, which will require that users present their credentials before being allowed to have any connectivity to the external public network(s). For this example, Aventail IPM is configured to use RFC1929 Username/Password for authenticating connections AutoSOCKS forwards to it. For additional information on how to configure the Aventail IPM product, consult the Aventail IPM *Administration Guide*.

Subsequently, in most Aventail IPM environments there are large numbers of clients that require installation and configuration. For completeness we will illustrate how to install and configure AutoSOCKS on a large number of client workstations. The easiest and best mechanism for installation of AutoSOCKS to many client workstations is to follow the AutoSOCKS network installation procedures. For our example, we will be installing the base AutoSOCKS client distribution to a network file server that will be used to pull the AutoSOCKS software and client configuration to the desktops. It is often the case that MIS personnel install single copies of AutoSOCKS for testing and evaluating prior to mass deployment. The configuration file that is created through the testing phases will then be copied to a shared file server for group access. This way each client workstation maintains the exact same configuration as determined by the network security policy.

Configuration Using Aventail VPN Server

The following example network configurations show the Aventail VPN Server configured for a Mobile VPN environment and a Partner VPN environment. This example emphasizes simplicity to facilitate easy adaptation to real world network designs.



Example Corporate Network Design using Mobile VPN

AutoSOCKS in an Aventail Mobile VPN Environment

The design used in the example above consists of two individual Ethernet segments, one public and one private. The public segment is used to host anonymous services available to the general public. The public access is provided through a router that is connected to the public Internet. The private segment is used to house all of the corporation's private network resources and data to be used only by internal company employees. The Aventail VPN Server depicted in this example is used to provide secure and monitored access to the private LAN for mobile employees and partners. For security reasons the Aventail VPN Server is configured such that operating system routing is disabled. Therefore, no direct network connections between the public LAN and the private LAN can be created without being securely proxied through the VPN server.

The mobile user workstations connected to the public Internet are the client workstations, onto which, AutoSOCKS will be deployed. Due to the routing restrictions described above, these clients will have no network access beyond the Aventail VPN Server unless they are running AutoSOCKS. Depending on the security policy and the Aventail VPN Server configuration, AutoSOCKS will automatically proxy their allowed application traffic into the private network. In this is situation, AutoSOCKS will forward traffic destined for the private internal network to the Aventail VPN Server. Then, based on the security policy, the Aventail VPN

Server will proxy mobile end user traffic into the private network but only to those resources allowed. The client workstations we focus on in this section are Microsoft Windows based PC's.

The Aventail VPN Server in our example, has two network adapters configured to use the internal IP address of 10.1.1.1 and an external address of 129.79.100.64. Since the internal network address space is part of the IANA reserved address space (per BCP RFC 1918) routing MUST be disabled on this host and routing advertisements for this internal network MUST NOT be propagated to the outside world. End user authentication and encryption has been enabled on the Aventail VPN Server, which will require all end users to use AutoSOCKS to enable authentication and encryption of their sessions before being allowed to have any connectivity to the internal private network(s). For this example, the Aventail VPN Server is configured to use SSL for encryption of all sessions. For additional information on how to configure the Aventail VPN Server product, consult the Aventail VPN Server *Administration Guide*.

Installation and use of AutoSOCKS for remote access purposes differs a bit from its installation and use with the Aventail IPM product. First, configuration files need to be kept locally on the end user workstation or laptop. This is due to the inability to have a shared file server that allows direct access outside the perimeter of the private network. Second, not all traffic is passed through to the Aventail VPN Server. Only traffic that is destined for the internal network is authenticated and encrypted, all other traffic passes through AutoSOCKS unchanged. For instance, browsing the Internet from the mobile user workstation occurs as if AutoSOCKS was not even running in the background. Large sites with many mobile users will want to setup an internal file server and perform a network installation for use by all of the mobile users to install and configure AutoSOCKS easily. For more information, consult the "Network Installation."

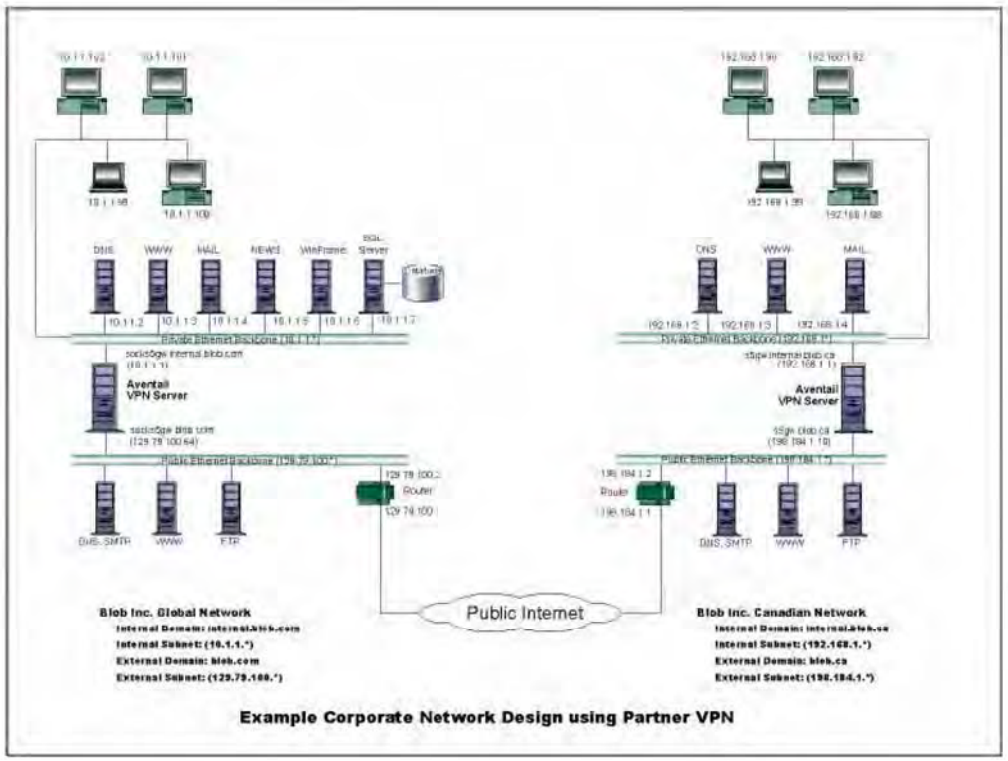


Figure 4. AutoSOCKS in a Partner VPN Environment

AutoSOCKS *Utilities Reference Guide*

Section II, the AutoSOCKS *Utilities Reference Guide*, covers the utilities available from the AutoSOCKS system menu. This section explains:

- Using commands in the System menu including Close, Hide Icon, Help, About, Credentials, Configuration File, Config Tool
- Using the Logging Tool to track AutoSOCKS activity and S5 Ping to check network connectivity

System Menu Commands

Even though AutoSOCKS requires little to no interaction with the end user, there are functions available by way of the AutoSOCKS System menu. To display the System menu, right-click the minimized AutoSOCKS icon (Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.1) or click the AutoSOCKS icon in the Taskbar tray (Windows 95 and Windows NT 4.0).

AutoSOCKS System Menu Commands

Menu Command	Function
Close	Closes AutoSOCKS.
Hide Icon	Hides the AutoSOCKS icon from view.
Help	Accesses online Help.
About	Displays Aventail AutoSOCKS About box.
Credentials	Displays authentication credentials.
Configuration File	Selects a new configuration file.
Config Tool	Runs the Config Tool.
Logging Tool	Runs the Logging Tool.
S5 Ping	Runs the ping and traceroute utilities.

Each of the commands are discussed in the paragraphs below.

Note: The Config Tool, Logging Tool, and S5 Ping commands are optional components and will only appear when they have been installed by the

network administrator. They are discussed in the sections "Logging Tool" and "S5 Ping" below.

Close

This command closes AutoSOCKS. Exiting AutoSOCKS may limit access to certain remote hosts or prevent you from using certain WinSock applications.

Hide Icon

This command hides the AutoSOCKS icon from view. AutoSOCKS will be running the background; however, the icon won't be visible in the system tray (Windows 95, Windows NT 4.0) or minimized on the desktop (for Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51).

Help

This command accesses AutoSOCKS online Help menu.

About

This command displays the Aventail AutoSOCKS About box which includes AutoSOCKS software copyright notification, version information, and so on. The **More** button displays a list of files used by the current version of AutoSOCKS.

Credentials

This command displays the Manage Credentials dialog box. Credentials include the information (such as username/password) that you enter when establishing a connection to a SOCKS server requiring user authentication. (AutoSOCKS prompts you with an authentication dialog box.) As long as your credentials are in memory, you can establish connections to associated SOCKS servers without needing to re-enter the authentication information.

Currently, there is no way to edit credential data fields; you can only delete the entire credential entry or clear the password portion of it. In either case, AutoSOCKS will prompt you to enter updated authentication information when you re-establish a connection to the associated SOCKS server.

Field	Definition
SOCKS Server	SOCKS server name
User Name	User name for the SOCKS server
Method	Numeric identifier of authentication method (2=username/password, 3=CHAP, 134=SSL)

To delete a credential entry

Delete authentication credentials when they are no longer correct. After the credentials are deleted, you'll be prompted to reenter them the next time you connect to the associated SOCKS server.

- Select the credential entry you wish to delete and click the **Delete** button.

  This deletes the credential information.

To exit the Manage Credentials dialog box

- Click the **OK** button to accept changes to the credentials and close the dialog box.

  -OR-

  Click the **Cancel** button to close the dialog box without accepting any changes you might have entered.

  Note: The **Apply** button makes changes permanent but keeps the dialog box open so you can keep working.

Configuration File

This command lets you load a different configuration file from the Select Configuration dialog box. AutoSOCKS defaults to AUTOSOCKS.CFG.

For more information about the configuration file, refer to "Creating Configuration Files."

To load a configuration file

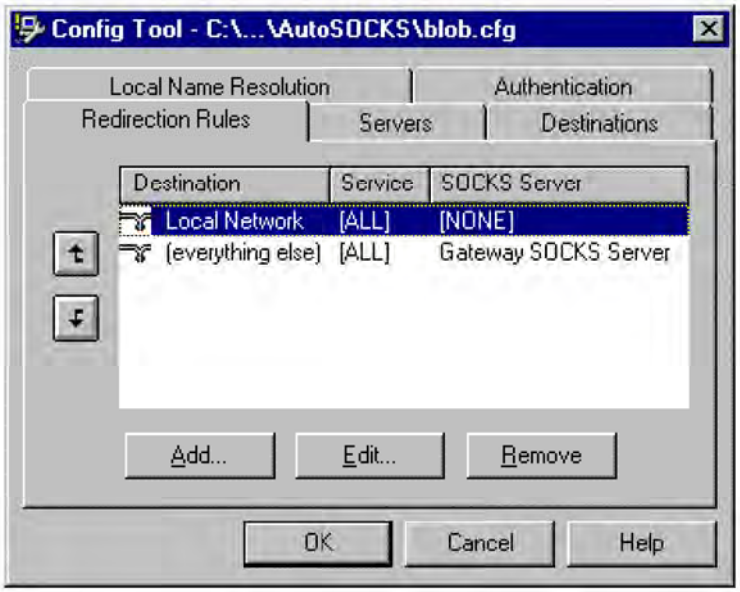
Check with the network administrator before making any changes to the configuration.

- Select the configuration file you wish to load and click the **Open** button.

  The new configuration file is transparently loaded into AutoSOCKS. AutoSOCKS must be restarted for the new configuration parameters to take effect.

Config Tool

The AutoSOCKS Config Tool creates configuration files used to determine how network requests should be routed and which authentication protocols should be enable. (This option may not be available to all users.)



Configuration files should be set up by a network administrator. They are usually created during AutoSOCKS installation but they can also be added, removed, or modified at any time. If necessary, several configuration files can be created for different users or user groups. Some configuration files may reside on a networked drive, accessible by multiple users; other configuration files may be tailored to a specific user on an individual workstation. The Config Tool dialog box is discussed in detail under "Creating Configuration Files."

Logging Tool

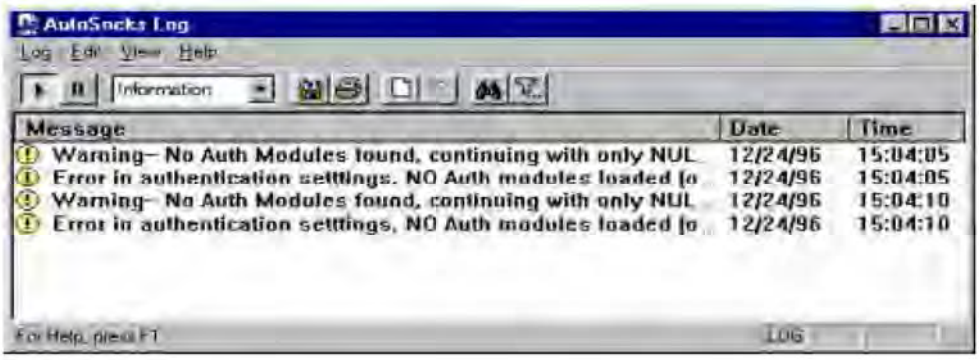
The Logging Tool is a diagnostic utility used to trace AutoSOCKS activity. (This option may not be available to all users.) When running a trace, the Logging Tool displays errors, warnings, and information messages as AutoSOCKS generates them. If desired, the message list can be saved to a log file for later study. Log files can be used to troubleshoot technical problems. They are also useful when running AutoSOCKS for the first time to ensure that network traffic is being routed appropriately.

To trace AutoSOCKS activity

1. Windows 95 or Windows NT 4.0: From the Programs command in the Start menu, point to Aventail AutoSOCKS and click Logging Tool.

   -OR-

   for Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51: From Aventail AutoSOCKS program group, double-click the Logging Tool program icon.



2. In the Log menu, select **Level** and then click one of the three levels of information you wish to trace.

   -OR-

   Select one of the three levels from the list on the toolbar.

Choose	To Log
Errors	Errors only
Warnings	Errors and warnings only
Information	Errors, warnings, and information

3. In the Log menu, click **Trace.**

   -OR-

   Click the **Trace On** button on the toolbar.

   The log window will now record and display trace information as it is generated by AutoSOCKS. You can tell when the trace function is active because messages are scrolling down the screen and the **Trace On** button is depressed.

5. When you're ready to stop the Trace function, click **Trace** in the Log menu

   -OR-

   Click the **Trace Off** button on the toolbar.

   The Trace function is stopped. You can now scroll through the results, print them, and/or save them to a file.

To save a log file

The Logging Tool allows you to append each new message to the end of a .LOG file as the trace is executed, or save the contents of the log window at any time. If you save as the trace is being executed, AutoSOCKS will append messages to the log file until you stop the log function. Data in the log window will not be retained unless it is saved.

There is no way to open a log file from within the log window. You must open a log file using a text editor such as Notepad.

- To save a log file as the data is being generated, click **Log to File** in the log menu. Enter the filename in the Select Log File dialog box.

  -OR-

  Click the **File Logging** button on the toolbar. Enter the filename in the Select Log File dialog box.

- To save the contents of the log window at any time, click **Save As** in the log menu and enter the filename.

To filter messages in the log window

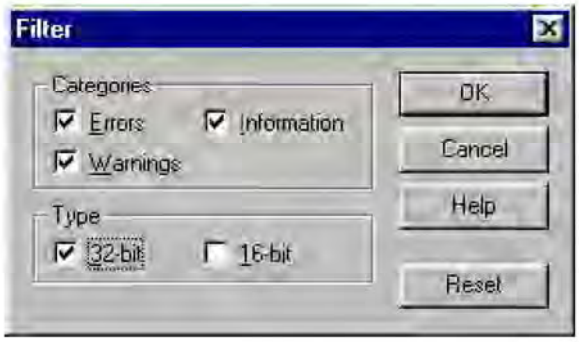
The contents of a log window can be filtered by selecting the types of messages you wish to view. Selecting a specific type of message can make it easier to scan the information onscreen. If the data has been saved to a log file, a view filter will not affect the file contents; it merely adjusts the screen display of those contents.

1. In the View menu, click **Filter Messages** to display the Filter dialog box

   -OR-

   Click the **Filter** button on the toolbar.

   Note: The Filter option is an on/off toggle. If the filter is enabled, click **Filter Messages** to turn it off, then select it again to display the Filter dialog box.



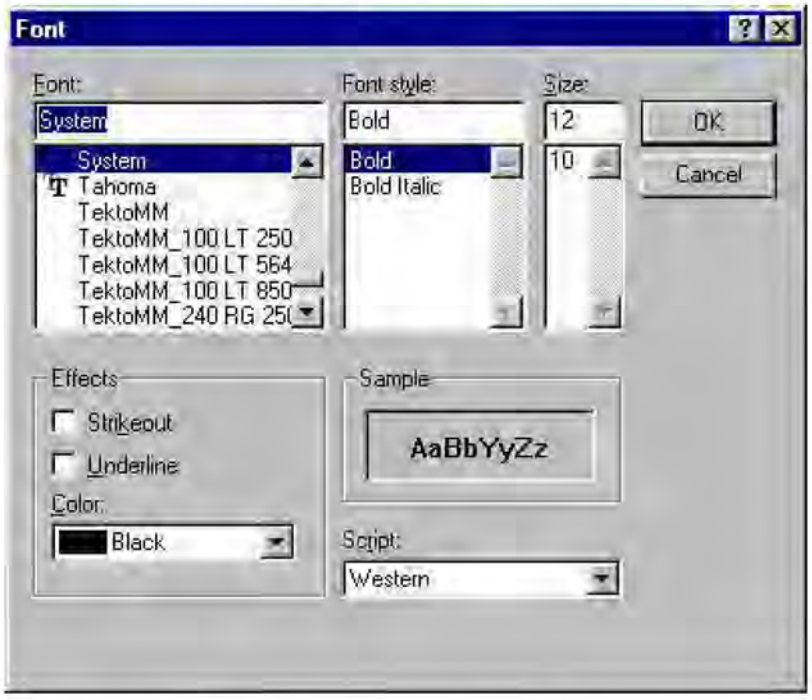
Field	Definition	
Categories	Select any of the three filters to display errors, warnings, and/or information in the log window.	
Type*	32-bit:	Show messages from 32-bit applications.
	16-bit:	Show messages from 16-bit applications.
	*These options are disabled if you're running 16-bit Windows.	

2. Under Categories, select one or more the three filter check boxes. The Log window will adjust the display based on your selection(s).
3. Under Type, select one or both of the check boxes.

To change the view parameters

The display font and window options can be customized as follows:

- In the View menu, click **Font**. Enter your font preferences into the standard Windows Font dialog box.



- To display and hide the toolbar and status bar, click **Toolbar** and/or **Status Bar** in the View menu.

To copy the log window

The log window contents can be copied to the Windows Clipboard.

- To copy all of the window contents to the Windows Clipboard, click **Select All** in the View menu. Then click **Copy** in the Edit menu or click the **Copy** button on the toolbar.
- To copy selected messages to the Windows Clipboard, drag the mouse over the messages to highlight them. Then click **Copy** in the Edit menu or click the **Copy** button on the toolbar.

To print the log window

The contents of the log window can only be printed in its entirety.

- To print the log window contents, click **Print** in the log menu.

  -OR-

  Click the **Print** button on the toolbar.

  The entire contents of the window will be printed, regardless of whether you have specific messages selected. If the display has been filtered, only the filtered messages will be printed.

To find a specific message

The Find function will only work with data displayed in the window. If the display has been filtered, only the filtered messages will be searched. The Find dialog box remains active until you close it.

- In the Edit menu, select **Find.**

  -OR-

  Click the **Find** button on the toolbar.

  Then enter your search parameters into the Find dialog box.

To clear the log window

Log window contents should be cleared when you're ready to execute a new trace, and you no longer need to see the old data.

- In the Edit menu, select **Clear All**.

  -OR-

  Click the **Clear All** button on the toolbar.

To close the log window

When you're ready to close the Log window, make sure you've saved the contents of the trace for later reference if necessary. All settings are saved when you exit.

- In the File menu, select **Exit.**

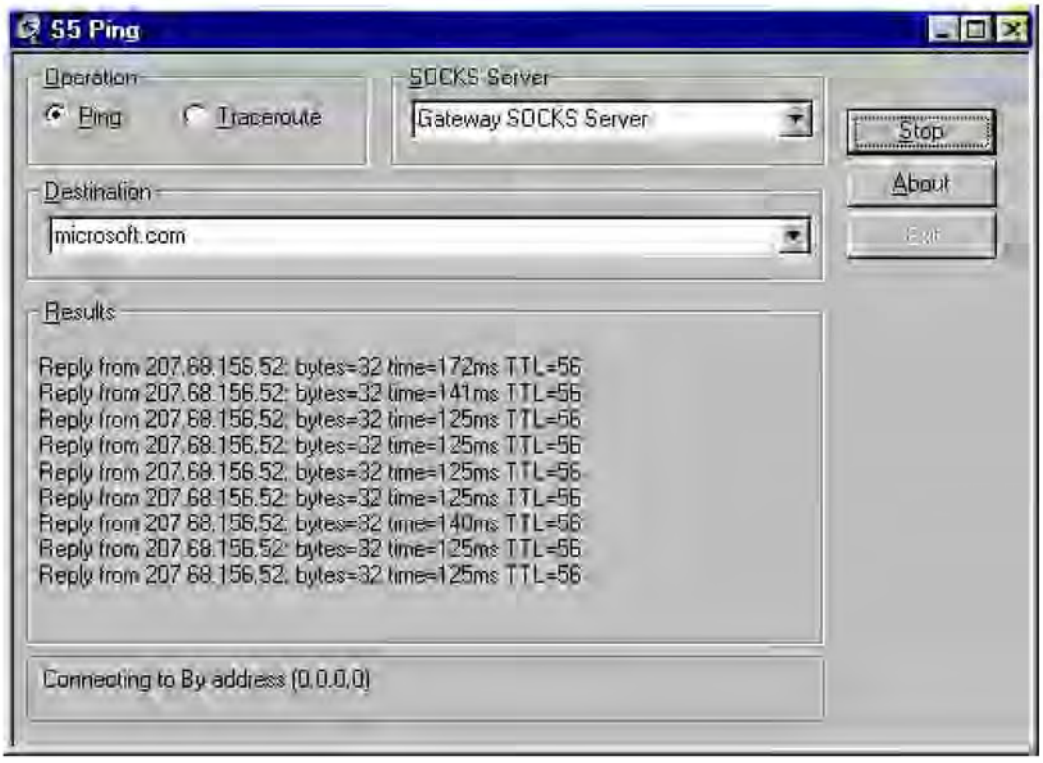
S5 Ping

Two of the most useful diagnostic tools in an administrator's arsenal are ping and traceroute.

- The ping utility checks for network connectivity between two hosts and returns information about the quality of the connection.
- The Traceroute utility checks for network connectivity by displaying information about routers between two hosts. It displays information for each hop.

Ping and traceroute both use Internet Control Message Protocol (ICMP). SOCKS v5 is designed to handle TCP and UDP protocols; however, ICMP is not supported. Because ping and traceroute are based on ICMP, there's no way to directly proxy a ping or traceroute request. To circumvent this problem, AutoSOCKS provides a utility called S5 Ping.

S5 Ping will ping (or traceroute to) a host outside of a SOCKS server by having the client request the SOCKS v5 server to ping the host in question. When a response from the host is returned, the SOCKS server relays the data back to the client and displays it in the S5 Ping window.



Field	Definition
Operation	Select the program you wish to run.
SOCKS Server	The SOCKS server which will execute the operation. If AutoSOCKS is already configured, this list will be preloaded with SOCKS servers from the configuration file.
Destination	The SOCKS server you wish to ping (or traceroute). If AutoSOCKS is already configured, this list will be preloaded with single host destinations defined in the configuration file. (See "Configuring AutoSOCKS.")
Results	The results of the operation once the connection succeeds. The format of the results will vary based upon the SOCKS server platform.

S5 Ping can be used whether or not AutoSOCKS is running. However, if the server that you're connecting through requires authentication, AutoSOCKS must be loaded. The availability of S5 Ping is determined by the network administrator when AutoSOCKS is first installed. In some cases, the S5 Ping command won't appear on the AutoSOCKS System menu or in the program group.

To run ping or traceroute using S5 Ping:

1. Launch S5 Ping.
2. Select the network operation to use (ping or traceroute).
3. Choose which SOCKS server will carry out the ping or traceroute operation.
4. Select the host to ping or traceroute.
5. Click the **Start** button to start the operation.

These procedures are described in the text below.

To launch S5 Ping

S5 Ping can be used whether or not AutoSOCKS is running.

1. Windows 95 or Windows NT 4.0: From the Programs command in the Start menu, point to Aventail AutoSOCKS and click **S5 Ping**.

   -OR-

   Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51: From Aventail AutoSOCKS program group, double-click the S5 Ping program icon.

   -OR-

   If AutoSOCKS is already running, choose the S5 Ping menu item from the AutoSOCKS tray icon menu (Windows 95, Windows NT 4.0) or from the minimized AutoSOCKS

icon System menu (Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51).

The S5 Ping window appears.

Note: S5 Ping will function without a properly configured AutoSOCKS; however, the user will be required to type the information about the target SOCKS server and target host into the SOCKS Server and Destination text boxes.

Once the S5 Ping window opens, you can execute a ping or traceroute network operation.

To run ping or traceroute using S5 Ping

S5 Ping has two modes of operation: ping and traceroute.

1. Under Operation, select one of the two options, Ping or Traceroute.
2. Under SOCKS Server, select a SOCKS server to carry out the operation. If no servers are listed (because S5 Ping did not locate an AutoSOCKS configuration file), type the SOCKS server's hostname or IP address.
3. Under Destination, select a single host destination to ping or traceroute. If no hosts are listed (because S5 Ping did not locate an AutoSOCKS configuration file), type the hostname or IP address of the host you wish to ping or traceroute.
4. Click the **Start** button to execute the operation. The **Start** button then changes to **Stop**. Results from any previous operation are cleared from the window.
5. If the SOCKS server requires authentication, you may be prompted with a server certificate or required to enter a username and password. (For more information about server certificates and username/password authentication, see "Managing Authentication Modules" in the AutoSOCKS v2.1 *Administration and User's Guide*.)
6. Once the connection to the host has been made, the information returned from the server will be displayed in the Results window.

To stop ping or traceroute

- Click the **Stop** button.

  This stops the operation and changes the **Stop** button back to **Start**. The results of the operation remain displayed in the S5 Ping window.

To exit S5 Ping

- Click the **Exit** button.

  This clears the results and closes the S5 Ping window.

AutoSOCKS *User Supplement*

AutoSOCKS automatically routes appropriate network traffic from a WinSock-compatible TCP/IP application such as an e-mail program or a web browser to a SOCKS-based server. (WinSock is a Windows TCP/IP interface that connects a Windows PC to the Internet.) The SOCKS server then sends the traffic to the Internet or the network. Your network administrator defines sets of rules by which this message traffic is to be routed.

This AutoSOCKS *User Supplement* is designed to familiarize you with aspects of the AutoSOCKS interface. Because AutoSOCKS is designed to run transparently, in most cases you'll interact with AutoSOCKS only when it prompts you to enter authentication information for a connection to a secure SOCKS server on the Internet or corporate intranet. You may also occasionally need to start and exit AutoSOCKS although network administrators often configure it to run automatically at startup.

If you have questions about how AutoSOCKS is running on your system, contact your network administrator. Details about other AutoSOCKS commands and utilities are described in the AutoSOCKS v2.1 *Administration and User's Guide*. You might find the section, "Getting Started" to be helpful.

How to Start and Close AutoSOCKS

Because network administrators often set up AutoSOCKS to run minimized at startup, you may never need to actually launch the AutoSOCKS application. When AutoSOCKS is started, it loads a default configuration file, AUTOSOCKS.CFG. This file contains the rules AutoSOCKS uses to properly route network traffic to and from your individual workstation. Your network administrator will inform you if the configuration file name should be different.

Closing AutoSOCKS may limit access to certain remote hosts or prevent you from using certain WinSock applications. Before closing AutoSOCKS it's a good idea to check with your network administrator.

To start AutoSOCKS

- Windows 95 and Windows NT 4.0: From the Programs command in the Start menu, point to Aventail AutoSOCKS and click AutoSOCKS v2.1.

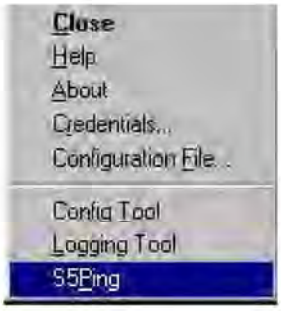
  -OR-

  Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51: In the Aventail AutoSOCKS program group, double-click the AutoSOCKS v2.1 program icon.

  You'll see a minimized AutoSOCKS icon indicating that AutoSOCKS is running in the background. In Windows 95 and Windows NT 4.0, this icon is located in the system tray on the Task bar.

To close AutoSOCKS

- Windows 95 and Windows NT 4.0: In the system tray, right-click the minimized AutoSOCKS icon to display the Aventail System menu, and click **Close**.



-OR-

Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51: Click the minimized AutoSOCKS icon to display the Windows System menu, and click **Close**.

Note: The Config Tool, Logging Tool, and S5Ping may not appear on the Aventail System menu in the program group. This is a configuration option determined when AutoSOCKS is first installed.

How to Enter Authentication Credentials

Some SOCKS servers ask you to authenticate yourself before you are allowed to access them. If you try to connect to a secure SOCKS server, AutoSOCKS may display a dialog box asking you to enter authentication credentials. (For some types of authentication methods, your input isn't required.) Credentials can be as simple as your username or password, or they can be more complicated information. Credentials are assigned to you by your network administrator.

Note: Never talk about credentials over cellular or cordless phones. These lines are not secure and you could be compromising system integrity. If you've mistakenly done so, be sure to let your network administrator know so that you can be assigned a new password.

Currently, AutoSOCKS supports four kinds of user authentication protocols: Username/Password, Challenge Handshake Authentication Protocol (CHAP), Secure Socket Layer (SSL), and SOCKS v4 Identification. To read more about these protocols, see "Managing Authentication Modules" in the AutoSOCKS v2.1 *Administration and User's Guide*.

Once you enter your credentials, AutoSOCKS will save them in memory. This is known as memory caching. Memory caching stores the credentials for the current session only. When

you restart AutoSOCKS or Windows, the memory cache is flushed. If you reconnect to the secure SOCKS server, you must again enter your credentials as prompted.

The following discussion includes Username/Password, CHAP, and SSL authentication. SOCKS v4 authentication does not require user interaction and therefore is not covered in this supplement.

Username/Password and CHAP Authentication

Username/Password and CHAP authentication use basically the same dialog boxes.

To enter authentication credentials

If the secure SOCKS server to which you're connecting uses Username/Password or CHAP authentication, you'll see a dialog box similar to the following:

Note: If you don't know what to enter into the dialog box fields, check with your network administrator.

1. In the Username text box, type your user name.

   Press TAB to move to the next field, or click the Password text box to place the insertion point. Be sure to type your username and password accurately.

2. In the Password text box, type your password.

   Your password is concealed as you type it; it displays on screen as a series of asterisk (*) characters.

3. Under Credential Caching, use the default option **Cache** for this session. Click **OK**.

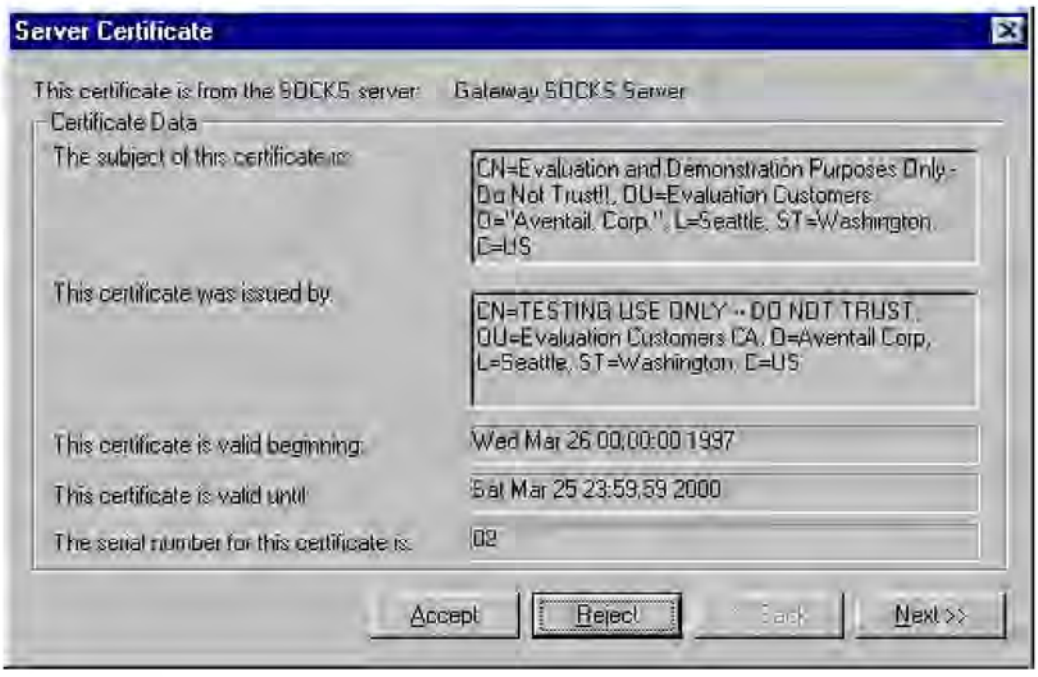
   When you click OK, your credentials are sent to the secure SOCKS server and if they are accepted, you'll continue your processing without hindrance.

If your credentials are refused by the server, the application will display an alert stating that the message traffic didn't go through. Try the transaction again, reentering your username/password. If problems persist, contact your network administrator.

SSL Authentication

SSL authentication, originally developed by Netscape for secure Web communications, uses *authentication certificates* to identify authorized users. A certificate is essentially an electronic "statement" which verifies the integrity of a connection. When you attempt to connect to an SSL server, AutoSOCKS may display the SSL certificate sent by the server. This may not always be the case, depending on how your network administrator has configured the system.

Note: It isn't the mission of this supplement to explain the intricacies of authentication or the components of SSL certificates. If you're interested in learning more about them, talk to your system administrator or read about them in the AutoSOCKS v2.1 *Administration and User's Guide* under "Managing Authentication Modules."



To accept an SSL certificate

Because anyone can issue a certificate that says anything, you should accept certificates only from trusted sources. Otherwise, the information you receive may be invalidated. If you have any concerns about whether or not to accept a certificate, talk with your network administrator.

1. When you see a trusted certificate display on screen, click **Accept**.

   If you click **Reject**, your connection won't be established. If you click **Next**, you see a second "page" of the certificate data with the same Accept and Reject buttons.

   If you click **Accept**, the certificate is accepted as valid and AutoSOCKS *may* display a Username/Password dialog box for you to fill in. The Username/Password dialog will only display if sub-authentication is being negotiated. With SSL authentication, the network administrator has the additional option of requiring you to perform a second (sub) level of authentication.

2. In the **Username** text box, type your user name.

   Press **TAB** to move to the next field, or click the Password text box to place the insertion point. Be sure to type your username and password accurately.

3. In the **Password** text box, type your password.

   Your password is concealed as you type it; it displays on screen as a series of asterisk (*) characters.

4. Under Credential Caching, use the default option **Cache** for this session. Click **OK**.

   When you click OK, your credentials are sent to the secure SOCKS server and if they are accepted, you'll continue your processing without hindrance.

Appendix I: Troubleshooting

AutoSOCKS-related problems tend to fall into four categories: Installation, Network Connectivity, Configuration, and Application and TCP/IP Stack Interoperability.

AutoSOCKS Installation Problems

When the instructions in Installing AutoSOCKS in the AutoSOCKS v2.1 *Administration and User's Guide* are followed, problems installing AutoSOCKS are rare. When they occur, they are often the result of:

Toolbars, virus-checking utilities, or other Windows applications running during the installation

If any of these are found to have been running during a failed installation, close them, uninstall AutoSOCKS, reboot, and then re-install AutoSOCKS, taking care to ensure that the toolbars, virus-checking utilities, or applications were not automatically restarted when the system was rebooted.

Insufficient RAM or free space on the volume to which AutoSOCKS is being installed

If either of these is suspected as the cause of a failed installation, increase the available resources according to the System Requirements of the AutoSOCKS v2.1 *Administration and User's Guide* and retry the installation.

Corrupted AutoSOCKS installation media or corrupted or incomplete FTP of AutoSOCKS self-extracting, executable installation file

If corrupted AutoSOCKS installation diskettes are suspected causes of a failed installation, contact Aventail Technical Support for assistance in determining whether the files on the diskettes may have been corrupted and whether replacement diskettes must be obtained from Aventail or your vendor.

If corrupted or incomplete FTP transfer of AutoSOCKS installation files obtained over the Internet is suspected, retry the transfer, taking care to ensure that the FTP client is in binary mode and confirm that the transfer completes normally. Contact Aventail Technical Support to confirm that the byte size of the transferred installation file is correct.

Installation to a workstation on which AutoSOCKS was running or from which a previous version of AutoSOCKS was not completely uninstalled

If either of these circumstances is suspected causes of a failed installation, contact Aventail Technical Support.

Installation script errors

AutoSOCKS is installed with InstallShield. If InstallShield reports errors during a failed installation, note the text of the error messages and the specific circumstances in which they occurred and contact Aventail Technical Support.

Network Connectivity Problems

Before AutoSOCKS can be used to successfully redirect WinSock application connections:

1. The workstation on which AutoSOCKS is installed must also have a properly installed, Winsock-compatible, TCP/IP stack running on it.

This installation can be confirmed by successfully pinging the IP address of the workstation, from the workstation itself, using a WinSock ping application. If this test fails, the failure must be corrected before AutoSOCKS can be tested and before Aventail Technical Support can provide assistance.

2. Basic TCP/IP network connectivity must exist between the client workstation on which AutoSOCKS is installed and the SOCKS server(s) to which it is configured to redirect connections.

This connectivity can be confirmed by successfully pinging the SOCKS server(s) by IP address, from the client workstation. If this test fails, the failure must be corrected before AutoSOCKS can be tested and before Aventail Technical Support can provide assistance.

3. Basic TCP/IP network connectivity must also exist between the SOCKS server(s) and the network host(s) to which the SOCKS server(s) are expected to proxy connections.

This connectivity can be confirmed by successfully pinging the network host(s), by IP address, from the SOCKS server(s). If this test fails, the failure must be corrected before AutoSOCKS can be tested and before Aventail Technical Support can provide assistance.

AutoSOCKS Configuration Problems

This section addresses troubleshooting of simple AutoSOCKS configuration problems. Troubleshooting of complex AutoSOCKS configuration problems is beyond the scope of this section.

It is easiest to troubleshoot AutoSOCKS configuration problems by creating and testing simple AutoSOCKS configuration files, such as those that may be created with the AutoSOCKS Configuration Wizard. However, all references to host and domain names should be removed from configuration files created with the wizard, before testing, to defer possible name resolution complications until the files can be demonstrated to work with IP addresses, alone.

Note: The IP address and SOCKS port number of the SOCKS server(s) to which AutoSOCKS must connect must be known, before troubleshooting AutoSOCKS configuration problems. Neither AutoSOCKS, nor Aventail

Technical Support, can discover the IP address or port number of the SOCKS server(s).

When troubleshooting AutoSOCKS configuration problems, confirm that the AutoSOCKS configuration file that is currently selected in the Configuration File… dialog is the one intended for testing.

After selecting a configuration file to test, open the AutoSOCKS Config Tool and:

1. Confirm that the SOCKS server has been correctly identified by IP address.

Click on the Servers tab, click on the server alias, and then click on the **Edit** button. Compare the IP address in the Hostname or IP: field with that of the SOCKS server.

If the SOCKS server is a SOCKS v5 server, click on the SOCKS v4 radio button in the SOCKS Version section of the Servers tab. Then click on the **Detect Version** button. The selection should revert to the SOCKS v5 radio button, indicating that AutoSOCKS detected a SOCKS v5 server running at the IP address specified in the Hostname or IP: field.

If, on the other hand, the SOCKS server is a SOCKS v4 server, click on the SOCKS v5 radio button in the SOCKS Version panel. Then click on the **Detect Version** button. The selection should revert to the SOCKS v4 radio button, indicating that AutoSOCKS detected a SOCKS v4 server running at the IP address specified in the Hostname or IP: field.

If **Detect Version** fails to detect a SOCKS server of either version, it is possible that no SOCKS server is running on the host identified in the Hostname or IP: field. Contact your SOCKS server administrator to confirm that the SOCKS server is running at the address specified.

2. Confirm that all AutoSOCKS Authentication Modules are enabled.

Click on the Authentication tab and confirm that the "traffic light" icons for all of the Authentication Modules are green, indicating that the modules are enabled. Enabling all the modules configures AutoSOCKS to attempt any form of authentication demanded by the SOCKS server or null (no) authentication. Note the form of authentication demanded by the SOCKS server and, if necessary, obtain the proper authentication credentials, such as a SOCKS server username and password, from the SOCKS server administrator.

3. Confirm that the network hosts to which the SOCKS server is expected to proxy connections are within a redirected destination.

Click on the Destinations tab, click on the Destination which includes the network host to which the SOCKS server is expected to proxy connections, and then click on the Edit button. Confirm that the definition of the Destination includes the network host.

Next, click on the Redirection Rules tab. Confirm that connections to the Destination are configured to be redirected by the SOCKS server.

After making any necessary changes to the AutoSOCKS configuration, restart AutoSOCKS and then restart any WinSock applications, before testing the new configuration.

Application and TCP/IP Stack Interoperability Problems

AutoSOCKS is intended to "automatically socksify" all "well-behaved" Winsock applications. Occasionally, Winsock applications are found which AutoSOCKS does not socksify, due to interoperability problems with the application.

AutoSOCKS is also intended to run on all WinSock-compliant Microsoft Windows TCP/IP stacks. Occasionally, WinSock stacks are found on which AutoSOCKS does not run as expected, due to interoperability problems with the stack.

If an application or stack inter-operability problem is suspected, report it to Aventail Technical Support. Aventail will make every effort to resolve interoperability problems.

AutoSOCKS Trace Logging

AutoSOCKS includes a Logging Tool for doing traces of AutoSOCKS and Winsock activity. AutoSOCKS traces are often useful in troubleshooting AutoSOCKS network, SOCKS server, and Winsock application interoperability problems. Aventail Technical Support engineers may request that you perform a debug-level trace, log it to file, and e-mail it to them.

Before Starting an AutoSOCKS Trace:

1. Close any WinSock applications that are running on the workstation.

2. Close AutoSOCKS, if it is running.

3. Start an AutoSOCKS Trace.

4. Click on the Windows Start | Programs | Aventail AutoSOCKS | Logging Tool menu bar item. The AutoSOCKS Logging Tool window should open, as illustrated in Figure 1, below.

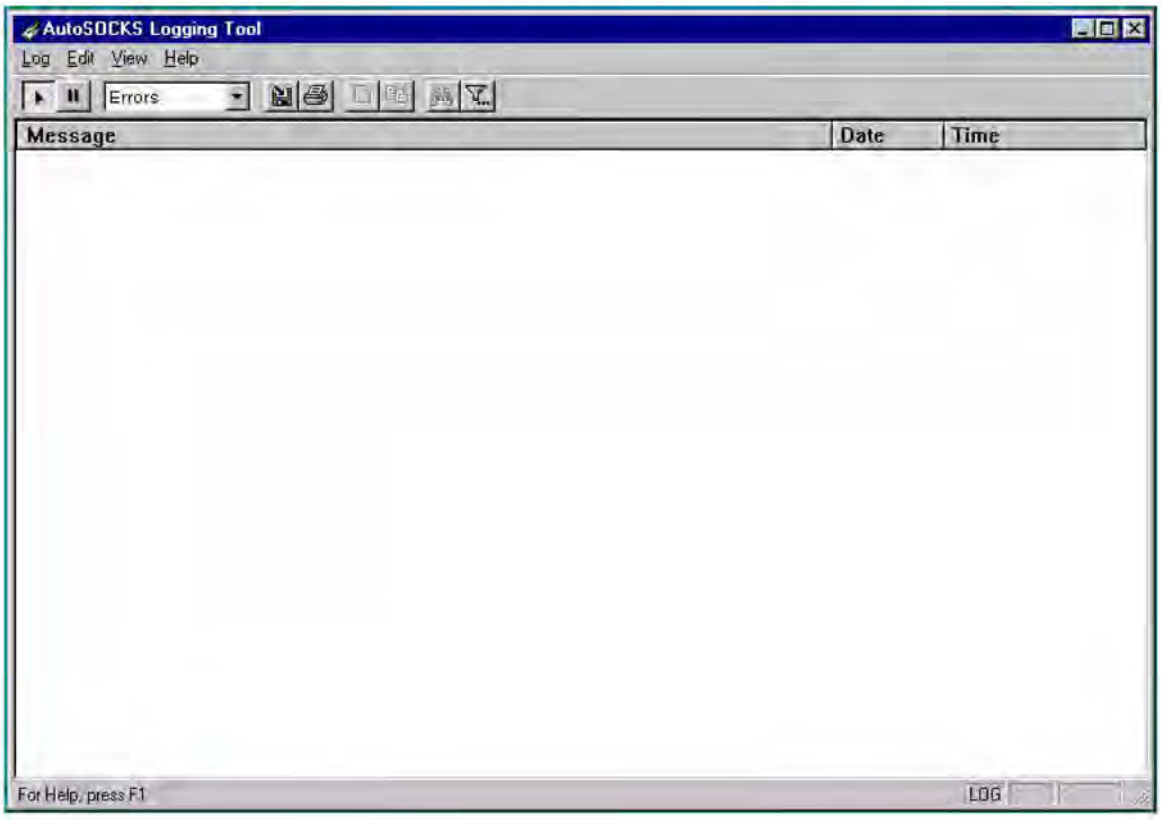


Figure 1

5. In the Logging Tool window Log menu, confirm that the Trace option is checked. If it is not, click on the Trace option, to check it.

Saving an AutoSOCKS Trace to a File:

1. In the AutoSOCKS Logging Tool window Log menu, confirm that the Log To File... option is checked. If it is not, click on the Log To File... option, to check it. The AutoSOCKS Logging Tool window Log menu should appear as illustrated in Figure 2, below.

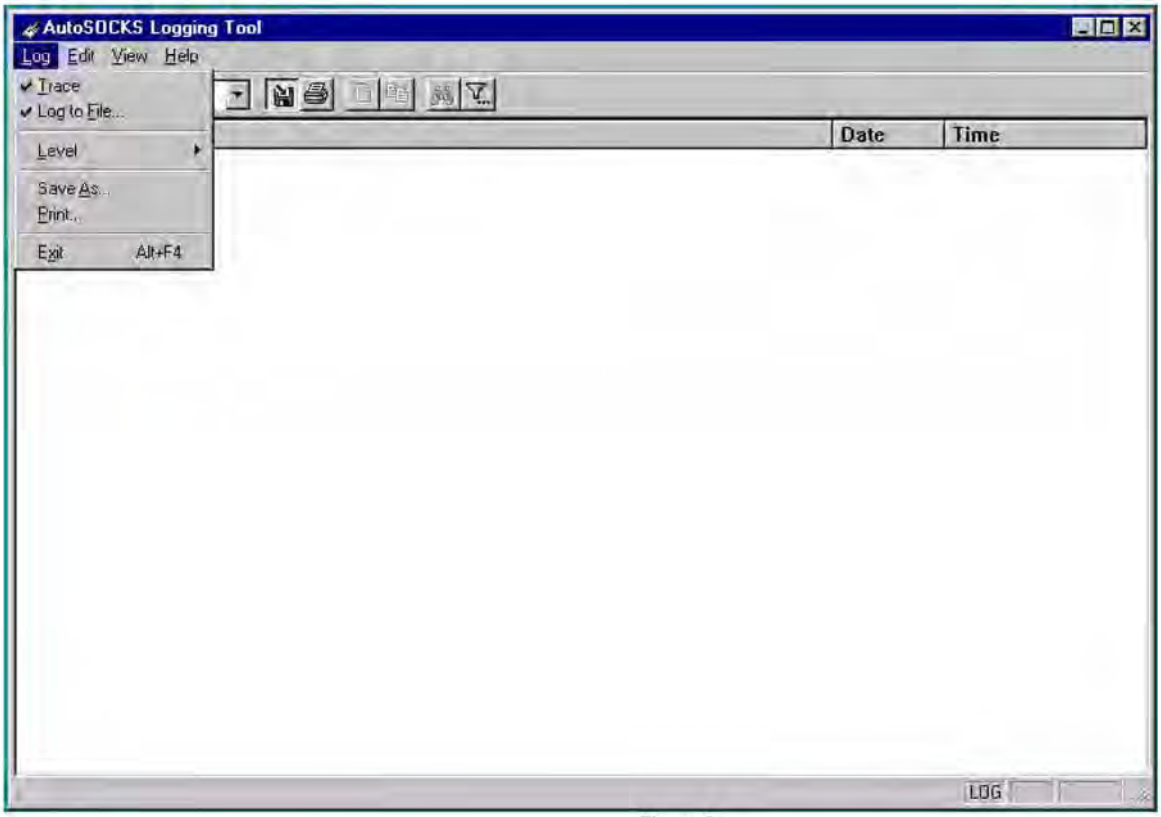


Figure 2

2. A Select Log File dialog box should appear, as illustrated in Figure 3, below. Enter a file name appropriate to later identify the file and click Save.

Figure 3

Setting the AutoSOCKS Trace Level to Debug:

1. Click on the AutoSOCKS Logging Tool window and then press <Ctrl><4>."Debug" should appear in the drop-down text box in the AutoSOCKS Logging Tool toolbar, as illustrated in Figure 4, below.

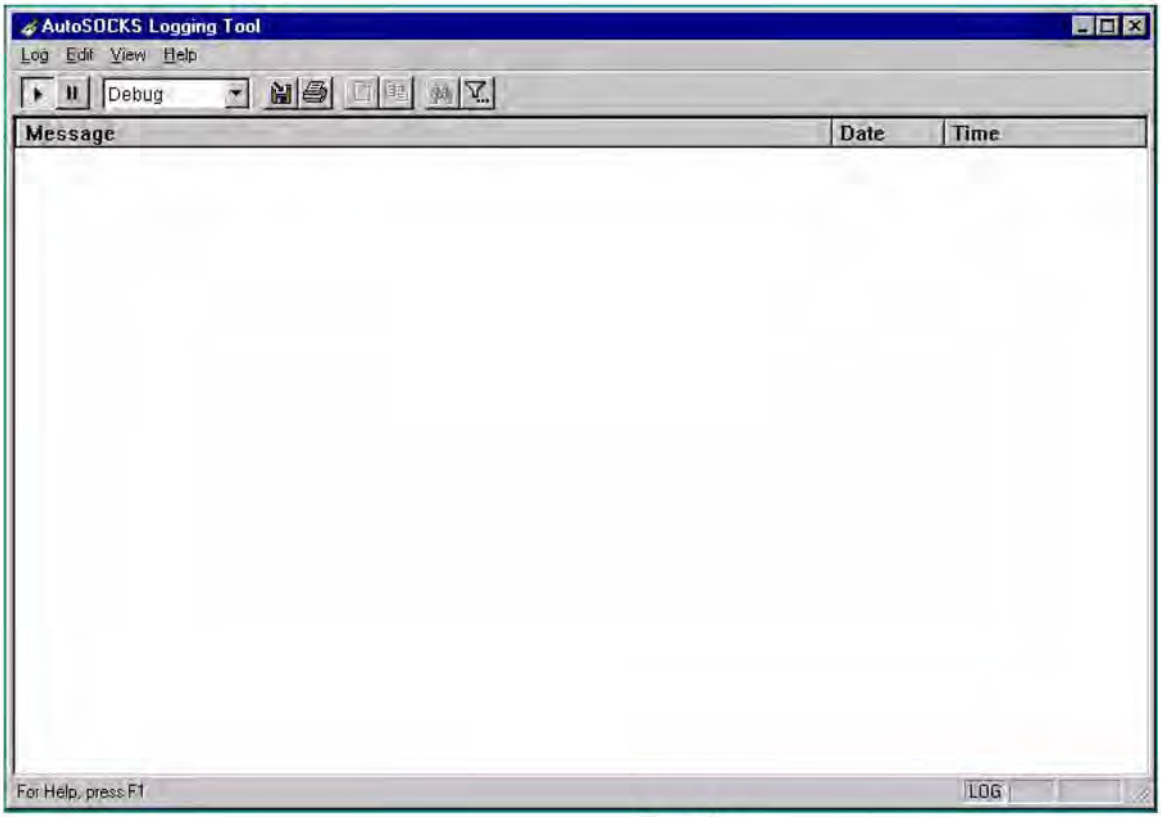


Figure 4

Note that, when tracing in Debug mode, not all messages that are displayed are indicative of error.

Logging Trace Data:

1. Start AutoSOCKS.
2. Start the Winsock application.
3. Reproduce the problem and only the problem.
4. Close the trace log file and confirm that it was saved.

Reporting AutoSOCKS Problems

Report AutoSOCKS problems to Aventail Technical Support, ideally by completing and submitting an AutoSOCKS Problem Report on the Support page of the Aventail website.

Glossary

alias
User-friendly name for destination network or host computer.

authentication
A method for identifying a user in order to establish access to a system resource or network. Authentication information such as username/password is entered via prompts.

certificate
A certificate is essentially an electronic "statement" which verifies that a certain RSA public key is associated with a particular name. Certificates are issued by a Certification Authority (CA).

client
A program or Internet service that sends commands to and receive information from a corresponding program known as a server. Most Internet services run as client/server programs.

configuration file
A file of information containing traffic redirection rules used to determine if and how SOCKS redirection should occur.

credentials
Credentials include the information (such as username/password) that you enter when establishing a connection to a SOCKS server requiring user authentication.

domain
Internet name for a network or computer system.

encryption
A security procedure that converts data into a format which can be read only by the intended recipient computer.

firewall
Software or hardware barriers that control the flow of information to Private networks.

host
A server connected to the Internet.

Internet Protocol (IP)
The basic data transfer protocol used for the Internet. Information such as the address of the sender and the recipient is inserted into an electronic "packet" which is then transmitted.

intranet
A network that is internal to a company or organization.

log window
The window of the Logging Tool which shows alerts, messages, and warnings generated by AutoSOCKS.

ping
A utility that determines if a remote host computer is up. ping sends data packets to the host. If the packets are not returned, the host is down.

protocol
Rules and procedures used to exchange information between networks and computer systems.

redirection rule
Rules defined in the configuration file which specify how network requests are routed to SOCKS servers.

server
A networked computer that shares resources with other computers. Servers "serve up" information to clients.

SOCKS
SOCKS is a security protocol. It acts as a proxy mechanism that manages the flow and security of data traffic to and from your local area network or intranet.

SSL
Security Sockets Layer, an authentication protocol.

Transmission Control Protocol (TCP)
A means of sending data over the Internet with guaranteed delivery.

Transmission Control Protocol/Internet Protocol (TCP/IP)
A suite of protocols the Internet uses to provide for services such as e-mail, ftp, and telnet.

traceroute
A utility that traces the routing of data over the Internet to a specific computer. Traceroute sends a data packet and then lists the intermediate host computers that it traverses on it's way to the destination machine.

User Datagram Protocol (UDP)
A means of sending data over the Internet without guaranteed delivery. Also known as "connectionless" protocol, it is used for data such as RealAudio®.

Universal Naming Convention (UNC)
A way of accessing a file or directory on another computer. For example:
//host/share/directory/file ("**share**" refers to the alias used to make the resource available.)

WinSock
(Windows Socket) A Windows component that connects a Windows PC to the Internet using TCP/IP.

workstation
Any computer connected to a network.

Index

EXHIBIT C

AVENTAIL CONNET V3.01/2.51 ADMINISTRATOR'S GUIDE

Aventail CONNECT

v3.01/v2.51

Administrator's Guide

Windows

Aventail

AVENTAIL CONNECT 3.01/2.51 ADMINISTRATOR'S GUIDE

808 Howell Street, Second Floor
Seattle, WA 98101
USA

http://www.aventail.com/

Printed in the United States of America.

TRADEMARKS AND COPYRIGHTS

Table of Contents

UTILITIES REFERENCE GUIDE

TROUBLESHOOTING

GLOSSARY

INDEX

Introduction

Welcome to the Aventail Connect 3.01/2.51 secure Windows client for 16- and 32-bit Windows applications. The client component of the Aventail ExtraNet Center, Aventail Connect is a secure proxy client based on SOCKS 5, the IETF standard for authenticated firewall traversal. Aventail Connect delivers enhanced security and simplifies SOCKS deployment for users and network managers.

Aventail Connect redirects WinSock calls and reroutes them based upon a set of routing directives (rules) assigned when Aventail Connect is configured. (For more information about WinSock, TCP/IP, and general network communications, see "Getting Started.")

On larger networks, Aventail Connect can address multiple SOCKS 5 servers based on end destination and type of service. This feature enables network administrators to effectively monitor and direct network traffic.

Aventail Connect is a proxy client, but when used with SSL it provides the ability to encrypt inbound or outbound information.

Features of Aventail Connect:

- Aventail Connect supports X.509 client certificates for strong authentication with SSL (when encryption is enabled)
- Automated Customizer utility simplifies client configuration, distribution, and installation
- SSL compression detects low bandwidth connections and compresses encrypted data (when encryption is enabled)
- Secure Extranet Explorer (via **Extranet Neighborhood** icon on desktop) allows users to securely access Windows or SMB hosts over an extranet connection (Windows 95, Windows 98, and Windows NT 4.0 only)
- Supports WinSock 2.0 (LSP) applications in Windows 98, and Windows NT 4.0, and WinSock 1.1 and WinSock 2.0 applications in Windows 95
- Supports WinSock 1.1 applications in Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51
- MultiProxy feature allows you to use a SOCKS server or an HTTP proxy to control outbound access
- Allows the use of port ranges for redirection rules
- Provides integration with SoftID™ and SecurID™ tokens
- Provides automated installation and uninstallation
- Credential cache timeout feature allows administrators to specify when credentials expire
- Provides optional password protection for configuration files
- Supports both SOCKS v4 and SOCKS v5 (RFC 1928 and RFC 1929) standards

- Enables network redirection through successive extranet (SOCKS) servers
- Includes a logging utility to troubleshoot problems with network connections
- Includes a Configuration wizard for simplified step-by-step creation of configuration files
- Allows internal network connections to pass through without interference
- Supports multiple authentication methods including SOCKS v4 identification, username/password, CHAP, CRAM, HTTP Basic (username/password), and SSL 3.0

SEE ALSO: *For more information on the differences between Aventail Connect 3.01 and Aventail Connect 2.51, see "What Does Aventail Connect Do?" in the Administrator's Guide.*

NOTE: *Not all versions of Aventail Connect have encryption enabled.*

About This Document

This *Administrator's Guide* provides basic information about Aventail Connect. It includes entry-level data for non-technical users, plus installation, setup, and configuration information for network administrators. This information is also available via Aventail Connect Help and the Aventail Web site at **http://www.aventail.com/content/products/docs/**.

Document Organization

This document is divided into three main sections: *Administrator's Guide*, *Utilities Reference Guide*, and *Troubleshooting*.

The *Administrator's Guide* describes procedures for setting up, installing, and configuring Aventail Connect for individual and multiple networked workstations. It also describes how to create a customized Aventail Connect package for distribution to multiple users.

The *Utilities Reference Guide* describes the Aventail Connect system menu commands and utility programs. It contains detailed information about using the S5 Ping utility and the Logging Tool, and documents the authentication/encryption modules and settings.

The document concludes with *Troubleshooting* and the *Glossary*.

You can also use the Quick Start Card, a short document designed to help you install Aventail Connect to an individual workstation, and the Aventail Connect flowchart, at **http://www.aventail.com/contents/solutions/presentations/quickstart/vpnclient.pdf**.

DOCUMENT CONVENTIONS

The following typographic conventions are used in this document. Exceptions may be made for online material; for instance, italics may be difficult to read online.

Convention	Usage
`Courier font`	Filenames, extensions, directory names, keynames, and pathnames. Command-line commands, options, and portions of syntax that must be typed exactly as shown.
Bold	Dialog box controls (**Edit...** buttons), e-mail addresses (**support@aventail.com**), URLs, (**www.aventail.com**), and IP addresses (**165.121.6.26**).
Italic	Placeholders that represent information the user must insert.

SEE ALSO: *A reference to additional useful information.*

NOTE: *Information the user should be aware of to increase understanding and/or efficiency of the software.*

CAUTION: *An operational item that the user should be aware of to avoid a network policy/software conflict, or lapse, which may create a MINOR security flaw.*

WARNING: *An operational item that the user should be aware of to avoid a network policy/software conflict, or lapse, which may create a SERIOUS security flaw.*

AVENTAIL TECHNICAL SUPPORT

Contact Aventail Technical Support if you have questions about installation, configuration, or general usage of Aventail Connect. Refer to the Aventail Support Web site, at **http://www.aventail.com/index.phtml/support/online_support.phtml**, or the Aventail Knowledge Base, at **http://www.aventail.com/index.phtml?page_id=03110000**, for the latest technical notes and information. Refer to the `readme.txt` documentation for additional information not included in the *Administrator's Guide*.

Aventail Technical Support:
Web site: **http://www.aventail.com/index.phtml/support/index.phtml**
E-mail: **support@aventail.com**
Phone: 206.215.0078
Fax: 206.215.1120

ABOUT AVENTAIL CORPORATION

Aventail Corporation is the leading vendor of extranet software. Its extranet solutions allow organizations to secure their networked communications and manage their employees' access to the Internet. Building an extranet gives organizations the ability to dynamically create a private communication or data channel over the Internet. Aventail's adherence to open security standards simplifies extranet deployment, enables interoperability, and leverages corporations' existing network investments. Its extranet solutions allow companies to extend the reach of their corporate extranets to customers, partners, remote offices, and worldwide employees.

Aventail Corporation
808 Howell Street, Second Floor
Seattle, WA 98101
Phone:206.215.1111
Fax:206.215.1120
http://www.aventail.com/
info@aventail.com

An aventail is a piece of chainmail armor worn around the neck area. In the 14th century, knights wore an aventail to protect themselves while in combat. Today, Aventail continues the tradition of protection by allowing organizations to securely communicate over the Internet.

Administrator's Guide

This section includes procedural and background information on installing Aventail Connect on both single and networked workstations. It includes:

- "Getting Started," with brief explanations of network security and communications
- Definitions of SOCKS and Aventail Connect
- Aventail Connect platform and installation requirements, with an introduction to WinSock 2.0 and LSP architecture
- "Installing Aventail Connect," which includes network diagrams of Aventail ExtraNet Center and SOCKS v4-based server configurations
- Directions on how to create and edit configuration files, and an introduction to the Aventail Customizer

NOTE: *Aventail understands the importance of a flexible, easy-to-use installation process. If you have feedback regarding the Aventail Connect installation procedures, or if there are additional features you want to see implemented, please e-mail comments to* **support@aventail.com**. *Your input is appreciated.*

GETTING STARTED

If you are new to Aventail Connect technology, the following section will help you understand what Aventail Connect is and does, and its relationship to network security in general.

NETWORK SECURITY IN A NUTSHELL

Escalating security threats are forcing companies to seek ways to safeguard their corporate networks and the information they exchange. The first response to these concerns has been the development of security firewalls—software barriers that control the flow of information. But firewalls are not designed to handle complex security issues, such as monitoring network usage, providing private communication over public networks, and enabling remote users to gain secure access to internal network resources.

Enter SOCKS v5, an Internet Engineering Task Force (IETF)-approved security protocol targeted at securely traversing corporate firewalls. SOCKS was originally developed in 1990, and is now maintained by NEC. SOCKS acts as a circuit-level proxy mechanism that manages the flow and security of data traffic to and from your local area network (LAN) or extranet. An application whose traffic

is proxied by SOCKS is considered "socksified." SOCKS is more than a standard security firewall. Other features:

- Client Authentication: (SOCKS v5 only) Authentication allows network managers to provide selected user access to internal and external areas of a network.
- Traffic Encryption: (SOCKS v5 only) Encryption ensures that network traffic is private and secure.
- UDP Support: (SOCKS v5 only) User Datagram Protocol (UDP) traffic has traditionally been difficult to proxy, with the exception of SOCKS v5.
- Aventail Connect supports X.509 client certificates within SSL: Includes a Certificate wizard for generating and processing client certificate requests.
- Cross-Platform Support: Unlike many other security solutions, SOCKS can be used on various platforms, such as Windows NT, Windows 95, Windows 98, and various forms of UNIX.

NOTE: *Not all versions of Aventail Connect include the SSL module for encryption.*

WHAT IS AVENTAIL CONNECT?

Aventail Connect is the client component of the Aventail ExtraNet Center. Aventail Connect works with the Aventail ExtraNet Server, the SOCKS 5 server component of the Aventail ExtraNet Center. You can use Aventail Connect as a simple proxy client for managed outbound access, and for secure inbound access.

Aventail Connect automates the "socksification" of Transmission Control Protocol/Internet Protocol (TCP/IP) client applications, making it simple for workstations to take advantage of the SOCKS v5 protocol. When you run Aventail Connect on your system, it automatically routes appropriate network traffic from a WinSock application to an extranet (SOCKS) server, or through successive servers. (WinSock is a Windows component that connects a Windows PC to the Internet using TCP/IP.) The SOCKS server then sends the traffic to the Internet or the external network. Network administrators can define a set of rules that route this traffic.

Aventail Connect is designed to run transparently on each workstation, without adding overhead to the user's desktop. In most cases, users will interact with Aventail Connect only when it prompts them to enter authentication credentials for a connection to a secure extranet (SOCKS) server. Users may also occasionally need to start and exit Aventail Connect, although network administrators often configure it to run automatically at startup. Aventail Connect does not require administrators to manually establish an encrypted tunnel; Aventail Connect can establish an encrypted tunnel automatically.

To understand Aventail Connect, you first need to understand a few basics of TCP/IP communications.

TCP/IP Communications

Windows TCP/IP networking applications (such as telnet, e-mail, Web browsers, and ftp) use WinSock (Windows Sockets) to gain access to networks or the Internet. WinSock is the core component of TCP/IP under Windows, and is the interface that most Windows applications use to communicate to TCP/IP.

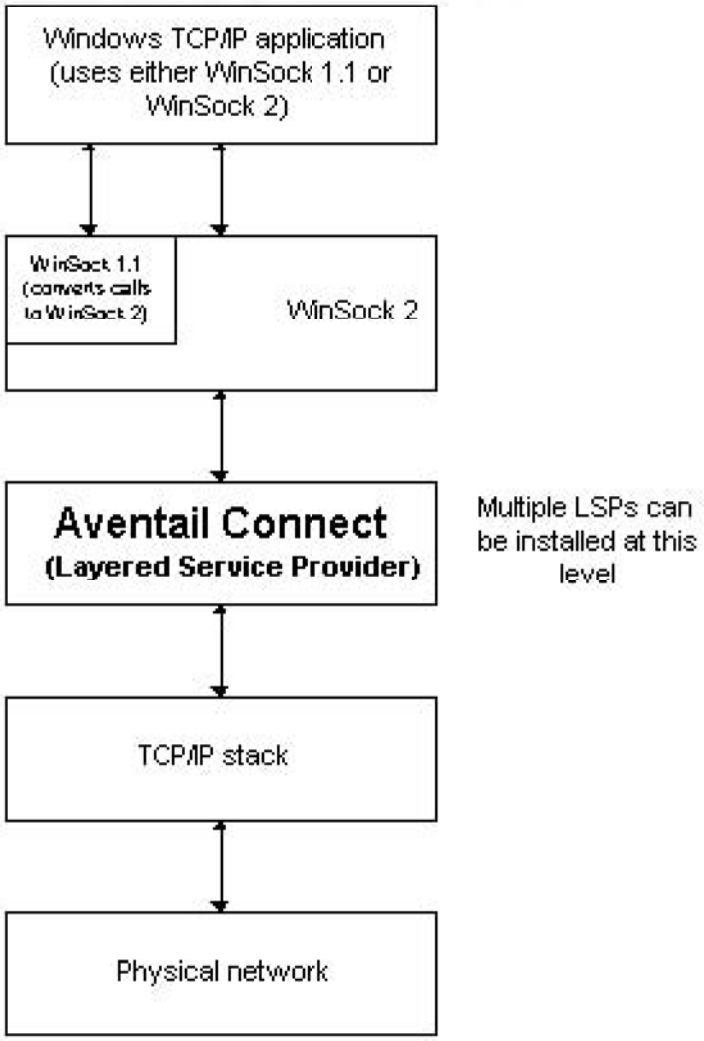
WinSock Connection to a Remote Host

Via WinSock, an application goes through the following steps to connect to a remote host on the Internet or corporate extranet:

1. The application executes a Domain Name System (DNS) lookup to convert the hostname into an Internet Protocol (IP) address. If the application already knows the IP address, this step is skipped.
2. The application requests a connection to the specified remote host. This causes the underlying stack to begin the TCP handshake, when two computers initiate communication with each other. When the handshake is complete, the application is notified that the connection is established, and data can then be transmitted and received.
3. The application sends and receives data.

WHAT DOES AVENTAIL CONNECT DO?

Aventail Connect slips in between WinSock and the underlying TCP/IP stack. (See diagram below.) As an application that sits between WinSock and the TCP/IP stack, Aventail Connect 3.01 is a Layered Service Provider (LSP). Aventail Connect can change data (compressing it or encrypting it, for example) before routing it to the TCP/IP stack for transport over the network. The routing is determined by the rules described in the configuration file.



Windows TCP/IP applications and Aventail Connect have no direct contact with one another; instead, each of them communicates through WinSock. Multiple LSP applications can be installed at the LSP level.

NOTE: *Aventail Connect does not alter or replace WinSock or any other core TCP/IP components (files) provided by the operating system.*

When the Aventail Connect LSP receives a connection request, it determines whether or not the connection needs to be redirected (to an Aventail ExtraNet Server) and/or encrypted (in SSL). When redirection and encryption are not necessary, Aventail Connect simply passes the connection request, and any subsequent transmitted data, to the TCP/IP stack.

The two most popular versions of WinSock are version 1.1 and version 2. Aventail Connect 3.01, like all LSPs, requires WinSock 2.0; WinSock 1.1 does not support LSPs. WinSock 2.0 includes backward-compatibility with all WinSock 1.1 applications. Not every platform supports WinSock 2.0 and its LSP structure.

- Windows 98 and Windows NT 4.0 support WinSock 2.0 natively. (Windows NT 4.0 requires Service Pack 3 or above, available from Microsoft.)
- Windows 95 supports WinSock 1.1. Windows 95 can also support WinSock 2.0, but you must install a Microsoft patch to add support for WinSock 2.0.
- Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51 do not support WinSock 2.0; they support only WinSock 1.1.

For those platforms that do not support WinSock 2.0 and LSP applications, Aventail includes Aventail Connect 2.51 on the Aventail Connect 3.01/2.51 CD. Aventail Connect 2.51 was designed for operating systems that support only WinSock 1.1. For Windows 3.1, Windows for Workgroups 3.11, or Windows NT 3.51 operating systems, setup will install Aventail Connect 2.51. If you are working on a Windows 95 operating system, setup will detect whether you have installed the Microsoft Windows 95 WinSock 2.0 Update. If setup detects the Microsoft update, which upgrades Windows 95 to support WinSock 2.0, setup will install Aventail Connect 3.01. If setup does not detect the Microsoft update, it will install Aventail Connect 2.51.

The Aventail Connect 2.51 user interface is identical to that of Aventail Connect 3.01; however, Aventail Connect 3.01 includes MultiProxy (see "Multiple Firewall Traversal"). Aventail Connect 2.51 does not include MultiProxy.

In the future, more Windows applications may require WinSock 2.0.

During installation, setup determines which version of Aventail Connect to install. On WinSock 2.0 platforms, Aventail Connect 3.01 is installed. On WinSock 1.1 platforms, Aventail Connect 2.51 is installed. The following table shows how setup determines which version of Aventail Connect to install.

Operating System	WinSock Support	Aventail Connect Version Installed
Windows 98, Windows NT 4.0	WinSock 2.0	Aventail Connect 3.01
Windows 95	With Microsoft patch: WinSock 2.0	Aventail Connect 3.01
	Without Microsoft patch: WinSock 1.1	Aventail Connect 2.51
Windows 3.1, Windows for Workgroups 3.11, Windows NT 3.51	WinSock 1.1	Aventail Connect 2.51

You can create custom packages that include one or both versions of Aventail Connect (3.01 and 2.51) Setup will determine which version to install on each workstation. (For more information, see "Customizer.")

WINDOWS 95 AND WINSOCK

The Microsoft Windows 95 WinSock 2.0 Update upgrades WinSock 1.1 to WinSock 2.0 in Windows 95. This patch (filename `w95ws2setup.exe`) is available from the Microsoft Web site, at **http://www.microsoft.com/windows/downloads/contents/Updates/W95Sockets2/default.asp**. Unless you need specific Aventail Connect 3.01 features, Aventail recommends that you do not upgrade from WinSock 1.1 to WinSock 2.0. If you do not upgrade to WinSock 2.0, Aventail Connect 2.51 will be installed.

If you do need to install the Microsoft Windows 95 WinSock 2.0 Update, follow the instructions provided by Microsoft. Reboot your computer after upgrading, prior to installing Aventail Connect.

HOW DOES AVENTAIL CONNECT WORK?

The following three steps are identical to standard WinSock communications steps described above; however, nested inside them are additional actions and options introduced by Aventail Connect.

1. The application does a DNS lookup to convert the hostname to an IP address. If the application already knows the IP address, this entire step is skipped. Otherwise, Aventail Connect does the following:
   - If the hostname matches a local domain string or does not match a redirection rule, Aventail Connect passes the name resolution query through to the TCP/IP stack on the local workstation. The TCP/IP stack performs the lookup as if Aventail Connect were not running.
   - If the destination hostname matches a redirection rule domain name (i.e., the host is part of a domain we are proxying traffic to) then Aventail Connect creates a false DNS entry (HOSTENT) that it can recognize

during the connection request. Aventail Connect will forward the hostname to the extranet (SOCKS) server in step 2 and the SOCKS server performs the hostname resolution.

- If the DNS proxy option is enabled and the domain cannot be looked up directly, Aventail Connect creates a fake DNS entry that it can recognize later, and returns this to the calling application. The false entry tells Aventail Connect that the DNS lookup must be proxied, and that it must send the fully qualified hostname to the SOCKS server with the SOCKS connection request.

2. The application requests a connection to the remote host. This causes the underlying stack to begin the TCP handshake. When the handshake is complete, the application is notified that the connection is established and that data may now be transmitted and received. Aventail Connect does the following:

   a. Aventail Connect checks the connection request.

      - If the request contains a false DNS entry (from step 1), it will be proxied.
      - If the request contains a routable IP address, and the rules in the configuration file say it must be proxied, Aventail Connect will call WinSock to begin the TCP handshake with the server designated in the configuration file.
      - If the request contains a real IP address and the configuration file rule says that it does not need to be proxied, the request will be passed to WinSock and processing jumps to step 3 as if Aventail Connect were not running.

   b. When the connection is completed, Aventail Connect begins the SOCKS negotiation.

      - It sends the list of authentication methods enabled in the configuration file.
      - Once the server selects an authentication method, Aventail Connect executes the specified authentication processing.
      - It then sends the proxy request to the extranet (SOCKS) server. This includes either the IP address provided by the application or the DNS entry (hostname) provided in step 1.

   c. When the SOCKS negotiation is completed, Aventail Connect notifies the application. From the application's point of view, the entire SOCKS negotiation, including the authentication negotiation, is merely the TCP handshaking.

3 The application transmits and receives data.

   If an encryption module is enabled and selected by the SOCKS server, Aventail Connect encrypts the data on its way to the server on behalf of the application. If data is being returned, Aventail Connect decrypts it so that the application sees cleartext data.

AVENTAIL CONNECT PLATFORM REQUIREMENTS

The following table lists the minimum system requirements for each of the platforms that Aventail Connect supports.

Platform	Processor	RAM	Extranet (SOCKS) Server
Windows 98; Windows NT 4.0 (requires Microsoft Service Pack 3 or above)	x86-based or Pentium personal computer	16 MB	Network-accessible SOCKS v4 or v5 compliant server
Windows 95; Windows NT 3.51	x86-based or Pentium personal computer	8 MB	Network-accessible SOCKS v4 or v5 compliant server
Windows 3.1; Windows for Workgroups 3.11	x86-based or Pentium personal computer	4 MB	Network-accessible SOCKS v4 or v5 compliant server

Aventail Connect 3.01 runs on the following operating systems:

- Windows 98
- Windows NT 4.0 (with Service Pack 3 or above, available from Microsoft)
- Windows 95, with the Microsoft WinSock 2.0 update (To install Aventail Connect 3.01, you must upgrade Windows 95 with the Microsoft WinSock 2.0 update prior to Aventail Connect installation and setup. If you do not install the Microsoft patch, Aventail Connect 2.51 will be installed. For more information, see "What Does Aventail Connect Do?".)

Aventail Connect 2.51 runs on the following operating systems:

- Windows 3.1
- Windows for Workgroups 3.11
- Windows NT 3.51
- Windows 95, without the Microsoft WinSock 2.0 update (If you do not upgrade Windows 95 with the Microsoft WinSock 2.0 update, Aventail Connect 2.51 will be installed. For more information, see "What Does Aventail Connect Do?".)

NOTE: *A WinSock-compatible 16- or 32-bit TCP/IP application must be installed and configured prior to running Aventail Connect. This can be the Microsoft-provided TCP/IP stack or a third-party TCP/IP stack.*

INTERFACE FEATURES

The following table lists the interface features for each platform. Each of these features is discussed in greater detail later in the *Administrator's Guide*.

Platform	Start Aventail Connect	Display System Menu	Open Secure Extranet Explorer	View Program Icon	Hide Program Icon
Windows 95, Windows 98, Windows NT 4.0	Start\Programs \Aventail Connect menu	Right-click **Aventail Connect** icon in system tray	Double-click **Extranet Neighborhood** icon on desktop	In system tray	Not available
Windows 3.1, Windows for Workgroups 3.11, Windows NT 3.51	**Aventail Connect** icon in Aventail Connect program group window	Click **Aventail Connect** icon in Aventail Connect program group window	Not available	Minimized on desktop	Configure during setup

INSTALLATION SOURCE MEDIA

Regardless of platform, Aventail Connect can be delivered on CD or as a network-delivered, self-extracting archive file.

- **CD**: The CD contains the Aventail Connect setup program, `setup.exe`. The setup program allows for an administrative setup. It also contains the *Administrator's Guide* and the *User's Guide* in the `\docs` directory, formatted for Adobe® Acrobat Reader.
- **Network-delivered Source Media**: The network-delivered source media is a self-extracting archive containing the required disk/directory structure within the archive file. The executable automatically extracts the Aventail Connect installation files and initiates setup. The archive filename will be similar to `as30s.exe`. This archive, or package, will also be available on the CD (located in the Utilities directory) to be used with the Customizer application. For more information, see the "Customizer" section.

INSTALLING AVENTAIL CONNECT

After your Aventail ExtraNet Server is set up, Aventail Connect can be installed to a single workstation or to multiple networked workstations. In either case, you must perform an initial installation of the software and create one or more configuration (.cfg) files. This procedure is described under "Individual Installation." Once the initial installation is complete, you can then install to a series of networked computers using the instructions and information described under "Network Installation."

NOTE: *To install or uninstall Aventail Connect on Windows NT machines, you must have administrative privileges on the machine (but not necessarily on the domain).*

If you are upgrading from an earlier version of Aventail Connect (VPN Client or AutoSOCKS), the following message may appear on your screen if you install a custom setup package using Aventail Customizer. This is not an error message. If this message appears, click **OK** and reboot your computer.

CONFIGURATION FILES

Integral to the initial installation of Aventail Connect is deciding how SOCKS traffic will be redirected through the network. Network redirection rules (used to determine if and how SOCKS redirection will occur) are defined in the Aventail Connect configuration (.cfg) file. Configuration files are initially created at the end of the installation process; however, you can add, edit, and remove configuration files at any time using the Config Tool (in Windows 95, Windows 98, or Windows NT 4.0 via the **Aventail** icon in the system tray on the taskbar; in Windows 3.1, Windows for Workgroups 3.11, or Windows NT 3.51 via the Aventail Program Group). The process of creating one or more configuration files is described under "Configuring Aventail Connect."

If you are installing Aventail Connect on multiple networked workstations, refer to "Network Installation" to determine the best method for maintaining and distributing configuration files. You can then proceed through the initial installation. The Installation Wizard will guide you through the steps, culminating with the option to create a configuration file.

CUSTOMIZED CONFIGURATION AND DISTRIBUTION

The Aventail Customizer is a utility that allows network administrators to customize Aventail Connect installation packages for distribution to multiple client work-

stations. Giving network administrators control over how setup packages are configured eliminates the need for end users to make installation and setup decisions at their workstations. The installation package is a self-extracting executable file. You can customize this file by adding license file, configuration file, or setup information for different authentication and encryption policies to meet various client-access needs of individuals or workgroups. You can customize configurations for multiple users and then distribute the package, providing easy access, download, and installation for users. You can reconfigure the Aventail Connect installation package anytime your network topology or security profiles change.

For more information about the Aventail Customizer, see the "Customizer" section.

INDIVIDUAL INSTALLATION

Before running setup, close all open Windows applications.

To install Aventail Connect

1. Installation procedures vary slightly, depending on which media source you use:
   - If you are installing directly from CD-ROM, run `setup.exe` from the Aventail Connect directory.
   - If you are installing from a network-delivered self-extracting archive, simply execute the archive file. This will extract the installation files and automatically launch the setup program.

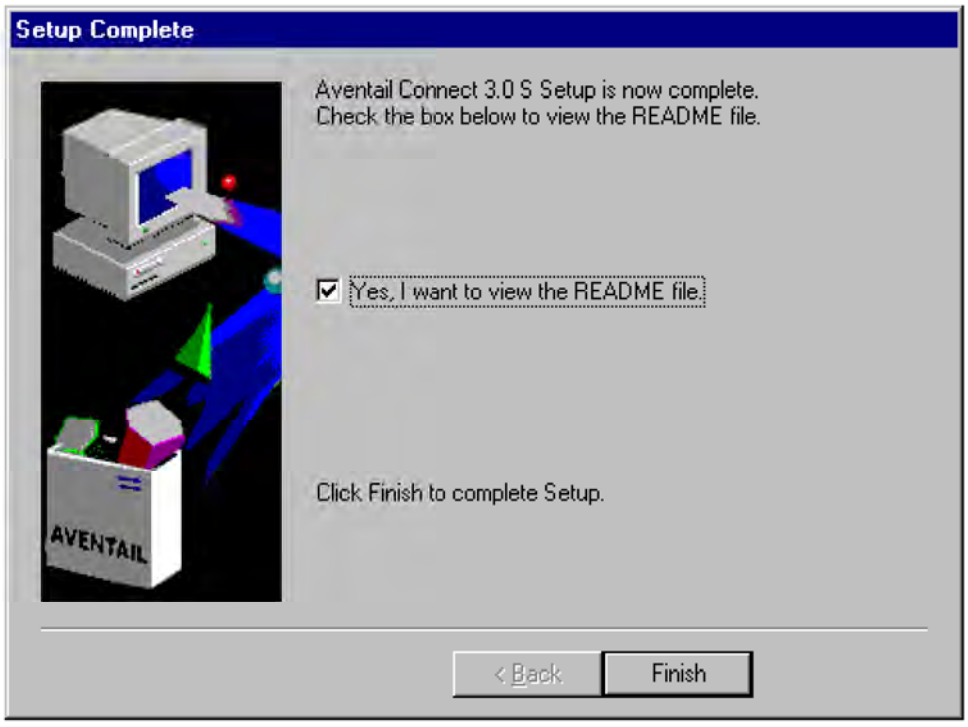
   The Aventail Connect installation wizard then guides you through the process of installing the Aventail Connect application.

NOTE: *You will be asked during the installation procedure if you would like Aventail Connect to be run automatically during startup. In most cases, you will select the* ***yes*** *option. Exceptions to this can be determined by the network administrator.*

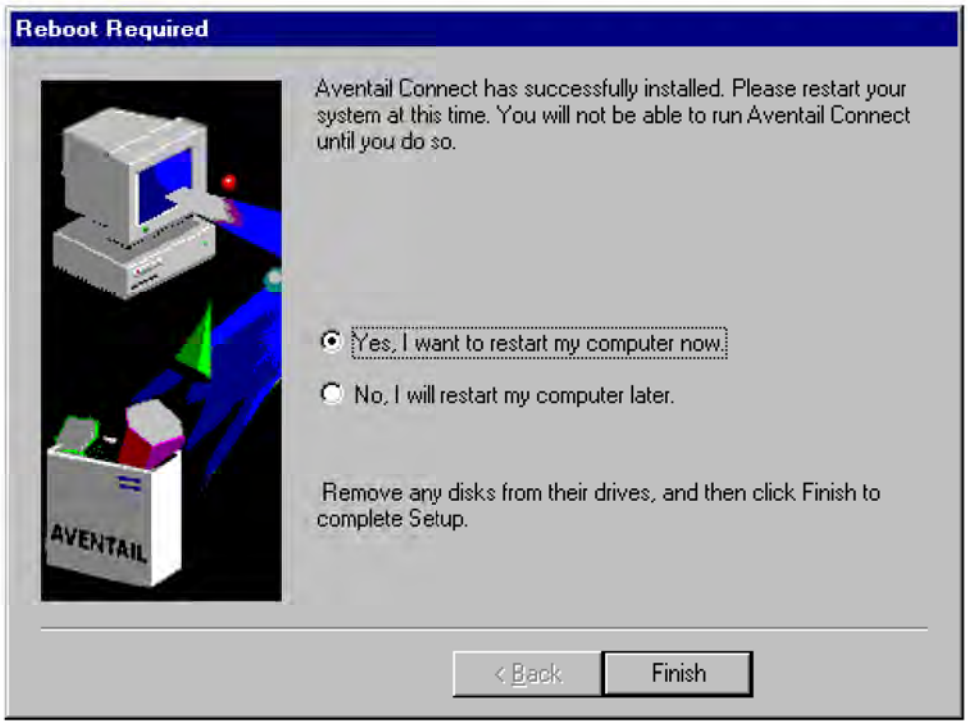
2. At the end of the setup program you can select the **Yes, I want to view the README file** box in the **Setup Complete** dialog box. This opens the `readme.txt` file, which contains the latest information on Aventail Connect.

-OR-

Simply click **Finish** to complete the setup program.



3. The setup program will then ask you if you want to restart your machine now or later.



4. After restarting your PC, Aventail Connect will launch automatically if, during installation, you chose "yes" when asked if Aventail Connect should be added to your startup directory. (If you selected the **no** option during installation, start Aventail Connect from the **Programs** menu.)
5. Aventail Connect will ask you if you want to run the configuration wizard.

   If you click **Yes,** then the configuration wizard will launch to help you create a new configuration file.

   If you click **No**, then Aventail Connect will ask you to select a configuration file.
6. After creating or selecting a configuration file, Aventail Connect will finish its installation procedure.

To uninstall Aventail Connect

The procedure to uninstall (remove) Aventail Connect varies depending on whether you are running a 16- or 32-bit Windows operating system.

- To uninstall Aventail Connect from Windows 95, Windows 98, and Windows NT 4.0, double-click **Add/Remove Programs** in the **Control Panel** window, click **Aventail Connect** on the list of programs on the **Install/Uninstall** tab, and click **Add/Remove**.
- To uninstall Aventail Connect on Windows 3.1, Windows for Workgroups 3.11, or Windows NT 3.51, use the **Uninstall** icon in the Aventail Connect program group.

NETWORK INSTALLATION

In general, the process of installing Aventail Connect to multiple networked workstations involves selecting a file server to use, creating a staging area for the Aventail Connect software, and placing the Aventail Connect package in a shared network directory or other publicly accessible location. Additional options include adding a default configuration file, license file, certificate and roots files, and SEEHosts files. You must place Aventail Connect files on a network drive that can be accessed as a mapped drive or, for Microsoft networks, via a UNC path name (`\\computer_name\share_name\Connect`).

An executable archive file (with a filename similar to `as30s.exe`) automatically extracts the Aventail Connect installation files and initiates setup. This archive, or package, is located in the Utilities directory of the CD and can be used in conjunction with the Customizer application. The package can also be manually configured to suit your network specifications. The default package includes all of the core Aventail Connect files, but does not include the custom network information.

NETWORKED CONFIGURATION FILE SETUP

There are a number of ways to set up networked client configuration files. These are the most common:

- Client configuration file shared on a mapped network drive (Novell or Microsoft)
- Client configuration file distributed via a Microsoft UNC path and file-name
- Local client configuration file common for all users, but distributed via an Aventail Connect package

ADMINISTRATOR-MAINTAINED SHARED CONFIGURATION FILES

This is the most desirable configuration method—multiple workstations sharing one or more administrator-maintained configuration files located in a common directory. The network administrator maintains the configuration file, and the administrator can quickly adapt any changes to network topology through a single configuration file. For example:

- A single networked (usually read-only) configuration file is shared by more than one client workstation. This method is appropriate when multiple workstations share identical traffic routing rules.
- Multiple configuration files are shared by multiple workstations. This option is useful when you have workstations organized into functional groups (engineering, marketing, accounting, etc.) with group-specific redirection rules.

SHARED CONFIGURATION FILE DISTRIBUTION

Shared configuration files can be easily distributed and, if necessary, updated via the network. Aventail recommends that you test all configuration files before distribution.

To distribute a shared configuration file

There are three methods for distributing shared configuration files.

- Copy the file to a Microsoft or Novell network drive accessible by all users. Make sure that users configure Aventail Connect to load the configuration file located on the mapped drive. You can preconfigure this information for users from a package install.

-OR-

- Copy the file to a Microsoft Windows workstation supporting UNC-sharing for file resources. (Both the 16- and 32-bit versions of Aventail Connect support specification of the configuration file using the Microsoft UNCs.) This distribution method has all the benefits of placing the file on a network mapped drive with the added bonus of convenience—users do not need to actually map the network drive.

-OR-

- Create a shared configuration file to be installed on workstations during the standard Aventail Connect installation/upgrade process. (You can build the configuration file into a package with Customizer.) Whenever Aventail Connect is installed or updated, it will automatically copy the shared configuration file to the user's workstation and set Aventail Connect to use it.

You can create and distribute shared configuration files with the Aventail Customizer. This automated wizard allows you to create custom setup packages for multiple users and then store the packages in a networked directory, providing easy access, download, and installation for users. You can include multiple local and/or remote configuration files.

ADMINISTRATIVE SETUP

There are two ways to install Aventail Connect: from the setup program (`setup.exe`), or from a setup package that you create using the Aventail Customizer. The setup program (`setup.exe`) allows you to manually install Aventail Connect. With the Aventail Connect setup package, you can select options that will customize setup based on your unique network environment. You can customize the setup package through the Customizer Editor or the Customizer Wizard. The Customizer *Editor* is a dialog box that allows you to manually enter or modify information about your custom installation package. The Customizer *Wizard* walks you through each step of creating a custom installation package. Aside from the user-interface differences, the Customizer Wizard and the Customizer Editor are identical. You can use both the Customizer Wizard and the Customizer Editor to create or modify a setup package. For example, you can create a package using the Customizer Wizard, then modify it with the Customizer Editor.

CUSTOMIZER

The Aventail Customizer simplifies and customizes the installation and setup process. Network administrators can reconfigure the self-extracting executable installation package (included in the Customizer directory of the distribution CD) to meet the various client-access needs of individuals or workgroups. Customizer offers a centralized approach to network configuration; network administrators may select the unattended setup mode, which eliminates the need for individual users to answer any setup configuration questions. Specifying unattended mode will cause the setup program to automatically install using default values for any options not explicitly specified.

The setup program (`setup.exe`) allows users to select any available setup options during installation of Aventail Connect. Customizer modifies the setup control file of a custom package; this file controls all of the settings within the setup package, before users receive the setup package. With a customized package, users will receive an installation package based on the administrator's defined settings.

As Customizer allows you to select various options to suit your setup and installation needs, the size of the setup package will vary, depending on which options you select. If size of the setup package is a concern, select setup options carefully to keep the package size manageable.

The Aventail Connect CD includes both versions of Aventail Connect (3.01 and 2.51). You can create custom packages that include one or both versions of Aventail Connect; setup will determine which version to install on each workstation. (For more information, see "What Does Aventail Connect Do?")

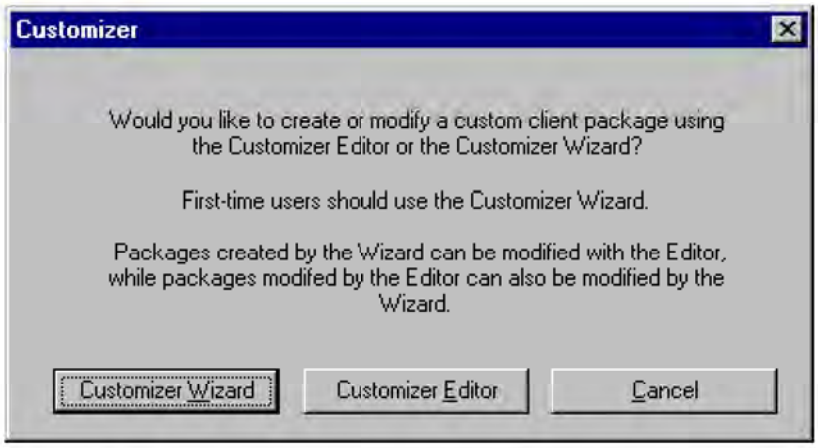
Aventail Connect requires a valid Aventail license file (`aventail.alf`) and one or more configuration (`.cfg`) files in order to function properly. Before installing Aventail Connect, make sure that users have these files. If users do not have a valid license file and/or configuration file(s), Aventail recommends that you include them in the installation package.

RUNNING CUSTOMIZER

The Customizer and the Aventail Connect installation package are included in the Customizer directory on the Aventail Connect CD. Before running Customizer, you must copy Customizer from the Aventail Connect CD to the local drive. You must also modify the Customizer attributes so it is not read-only.

To run Customizer, double-click the **Customizer** icon in the Customizer directory. To run Customizer from your hard drive, copy the Customizer and Aventail Connect directories into a common folder on the hard drive.

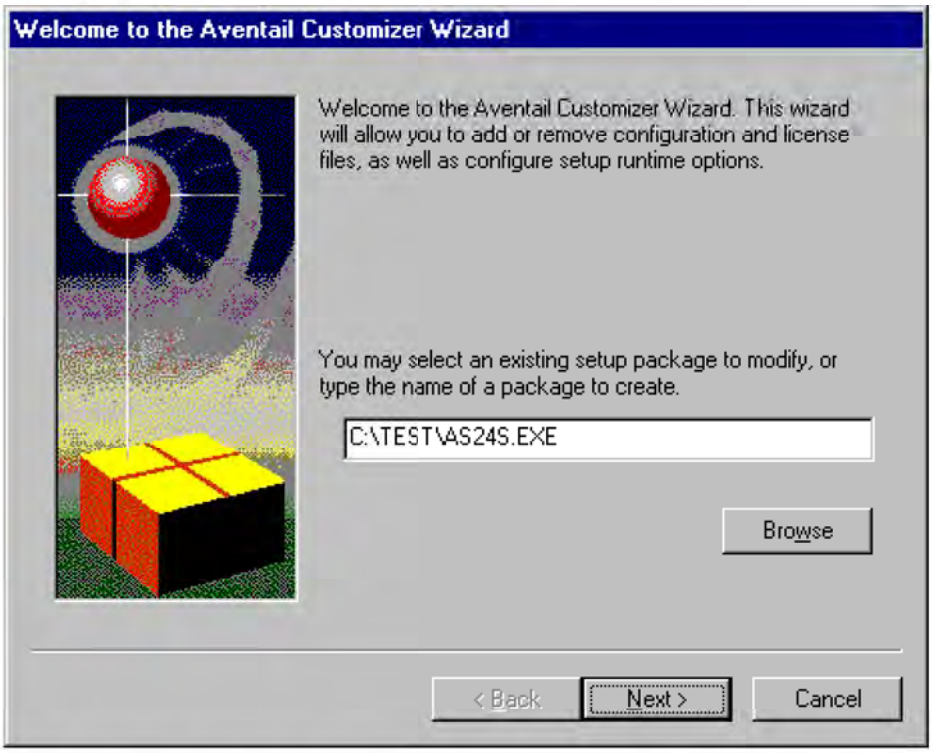
When you run Customizer, you will be prompted to select either the Customizer Wizard or the Customizer Editor.



- **Customizer Wizard:** This automated wizard walks you through the process of creating a new installation package or modifying an existing package. If you are unsure about which method to use, Aventail recommends that you use the Customizer Wizard.
- **Customizer Editor:** The Customizer Editor is a dialog box that allows you to manually enter information about the package you are creating or modifying.

CUSTOMIZER WIZARD

If you are using the Customizer Wizard to create a new setup package or modify an existing package, the Customizer Wizard will display a **Welcome...** screen, and will prompt you to enter the pathname of the package that you will be creating or modifying.



After you have specified the pathname of the package, the Customizer Wizard will prompt you to:

- Specify which platform(s) to support
- Add a license file, or leave an existing license file in the package
- Add or remove configuration files
- Select X.509 certificate files
- Select an extranet hosts (SEEHosts) file
- Specify a custom destination directory
- Specify whether or not to put program icons in a custom folder
- Enter command-line switches
- Specify whether or not to run setup in unattended mode
- Specify whether or not to add Aventail Connect to the startup directory
- Select any, all, or none of the following Aventail Connect components:
  - Extranet Neighborhood (Secure Extranet Explorer)
  - Configuration Tools (Config Tool and Configuration File command)
  - Diagnostic Tools (Logging Tool and S5 Ping)
  - Certificate Tools

- Install 32-bit support only (on Windows NT 3.51)

- Select any, all, or none of the following authentication modules:
  - SSL (Secure Sockets Layer)
  - CRAM (Challenge Response Authentication Method)
  - CHAP (Challenge Handshake Authentication Protocol)
  - UNPW (Username/Password)
  - SOCKS 4
  - HTTP Basic (username/password)
- Specify whether or not to run a command after Setup

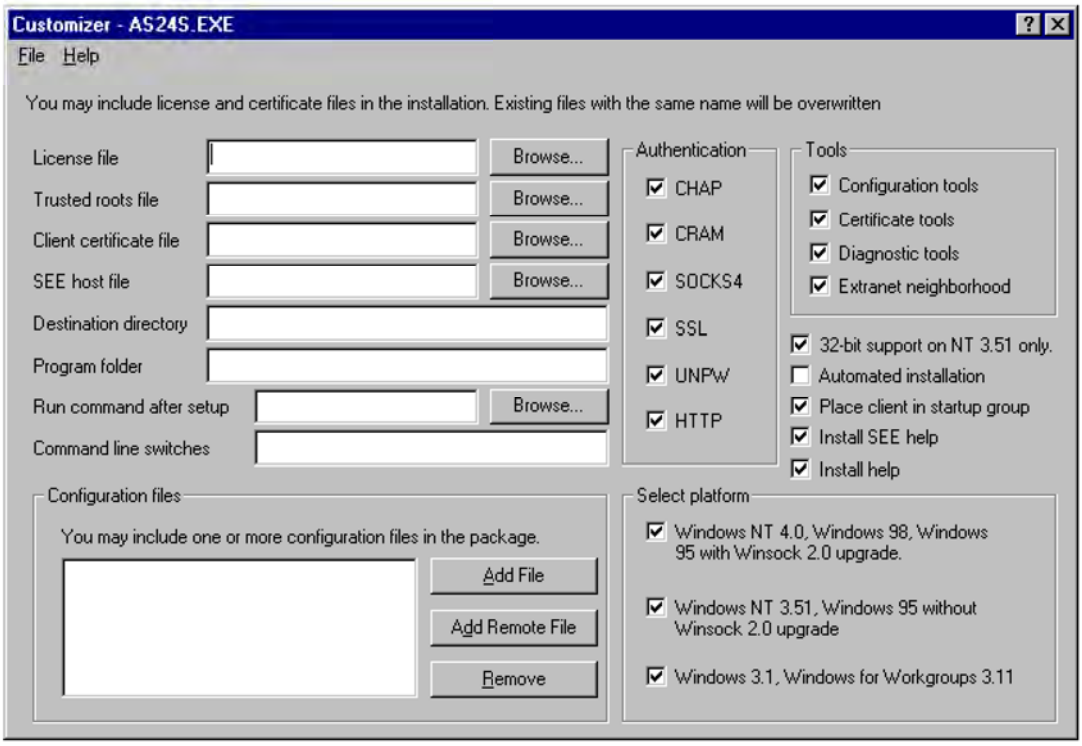
All of the features listed above are optional.

After entering or modifying the package information, the **Finished Installation Package** dialog box appears.

Clicking **Finish** saves your specifications and closes the Customizer Wizard. Clicking **Customize Further** allows you to view the **Customizer Editor** dialog box, where you can manually edit any of the information about your custom installation package.

CUSTOMIZER EDITOR

If you select the Customizer Editor as your tool to create a new setup package or modify an existing package, the **Customizer Editor** dialog box will appear. In this dialog box, you can manually enter or modify information about your custom installation package.



NOTE: *To view a list of tips on creating custom setup packages, click* ***Tips*** *on the* ***Help*** *menu in the* ***Customizer Editor*** *dialog box.*

After entering or editing your setup package information in the Customizer Editor, click **Save** (or **Save As**) on the **File** menu to save your changes. To close the **Customizer Editor** window, click **Exit** on the **File** menu.

The options in the Customizer Editor are identical to the options in the Customizer Wizard. These options are explained in the following paragraphs and tables.

Option	Settings	Default Setting
Pathname	Enter pathname	None
License file	Enter name of Aventail license file (must use `aventail.alf`)	None
Trusted roots file	Enter name of trusted roots file	None
Client certificate file	Enter name of file that contains certificate	None
Extranet (SEE) Hosts File	Enter name of extranet (SEE) hosts file	None
Destination directory	Enter name of destination directory	None
Program folder	Enter name of program folder	None
Run command after setup	Enter command to be run after setup	None
Command line switches	Enter command line switches	None
Configuration Files	Enter name(s) of local and/or remote configuration file(s) that Aventail Connect will use	None
Authentication Modules	SSL, CRAM, CHAP, UNPW, S4, or HTTP Basic	All
Tools	Configuration tools, Certificate tools, Diagnostic tools, or Extranet Neighborhood	All
32-bit support only, on Windows NT 3.51	Yes/No	Yes
Unattended Setup Mode/Automated installation	Yes/No	No
Add to Startup Directory	Yes/No	Yes
Install SEE help	Yes/No	Yes
Install help	Yes/No	Yes
Select platform	Windows NT 4.0, Windows 98, Windows 95 with WinSock 2.0 upgrade, Windows 95 without WinSock 2.0 upgrade, Windows NT 3.51, Windows 3.1, or Windows for Workgroups 3.11	All

The setup package options are discussed below.

- **Specify path for installation:** You can specify a path for installation, or you can select the default path. The default path for 32-bit operating systems is `c:\Program Files\Aventail\Connect`.

  For 16-bit-only operating systems, the default is `c:\Connect`.

NOTE: *If you are upgrading from an earlier version of Aventail Connect, Aventail Connect will install to the same directory that the earlier version of it was installed to.*

- **Platforms:** You must specify which operating systems need to be supported in the setup package. Aventail Connect 3.01 supports Windows 95 (with the Microsoft WinSock 2.0 update), Windows 98, and Windows NT 4.0 (with Service Pack 3 or above, available from Microsoft). Aventail Connect 2.51 supports Windows 95 (without the Microsoft WinSock 2.0 update), Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51. For more information, refer to "What Does Aventail Connect Do?"
- **Trusted Roots File and Certificate File:** If you want to use server certificates, you must include the trusted roots file that contains those certificates. If you want to use client certificates, you must specify the location of the file that contains the X.509 certificate.
- **Running Setup in Unattended Mode:** Unattended setup mode simplifies distribution of numerous client configuration files. The network administrator specifies all settings before users receive the Aventail Connect setup package file. No end-user input is required because the network administrator has already selected the setup options; users simply open the package file, which will automatically install on their workstations.

NOTE: *Specifying unattended setup mode will cause the setup package to automatically install using default values for any options not explicitly specified.*

- **Adding Aventail Connect to the Startup Directory:** If you choose to add Aventail Connect to the startup directory, Aventail Connect will automatically start when Windows starts.
- **Select Tools:** Aventail Connect gives you the option to install various components, including Extranet Neighborhood/Secure Extranet Explorer (SEE), configuration tools (Config Tool and Configuration File command), or diagnostic tools (Logging Tool and S5 Ping). The default value is to install all package components.
  - **Secure Extranet Explorer:** Secure Extranet Explorer (SEE) allows you to view your Extranet Neighborhood, which is accessed through

the **Extranet Neighborhood** icon on your desktop. Extranet Neighborhood functions much like Network Neighborhood, except Extranet Neighborhood allows you to browse, copy, move, and delete files from secured remote computers via an extranet, while Network Neighborhood displays all computers on your local network.

- **Config Tool:** The Aventail Connect Config Tool allows you to create configuration files that determine how network requests will be routed and which authentication protocols will be enabled. You can add, remove, or edit configuration files at any time. If necessary, you can create several configuration files for different users or user groups. If you want to prohibit end users from editing configuration files, do not include the Config Tool in the installation package.
- **S5 Ping:** S5 Ping allows you to use the ping and traceroute utilities, two diagnostic tools. The ping utility checks for network connectivity between two hosts and returns information about the quality of the connection. The traceroute utility checks for network connectivity by displaying information about routers between two hosts; it displays information for each hop.
- **Logging Tool:** The Logging Tool is a diagnostic utility that traces Aventail Connect activity. When running a trace, the Logging Tool displays errors, warnings, and information as Aventail Connect generates them. If necessary, the message list can be saved to a log file that can be used by Aventail Technical Support in troubleshooting technical problems. These traces are also useful when running Aventail Connect for the first time to ensure that network traffic is being routed appropriately.

- **Select Authentication Modules:** Aventail Connect lets you select any, all, or none of the following authentication modules: SSL, CRAM, CHAP, UN/PW, SOCKS v4, or HTTP Basic (username/password).
  - **Secure Sockets Layer:** Secure Sockets Layer (SSL) is a session-layer protocol for securing connections in a general, protocol-independent fashion.

NOTE: *In versions of Aventail Connect that do not include encryption, the Secure Sockets Layer (SSL) authentication module is not included.*

  - **CRAM:** The Challenge Response Authentication Method (CRAM) sends your username and password as clear text between extranet (SOCKS) servers, but encrypted between servers that support CRAM. Typically, CRAM subauthenticates within SSL, which provides both encryption and credential caching options.

NOTE: *In versions of Aventail Connect that do not include encryption, the CRAM authentication module is not included.*

- **CHAP:** The Challenge Handshake Authentication Protocol (CHAP) sends your username and password encrypted across the network to the destination server.
- **Username/Password:** The RFC 1928 (Internet standards document) Username/Password (UNPW) authentication protocol sends your username and password in clear text across the network to the destination server.
- **SOCKS 4 Identification:** Aventail Connect includes backward compatibility for the SOCKS 4 protocol. SOCKS 4 does not support password authentication, so only your username is sent, unencrypted, to the SOCKS server along with your connection request.
- **HTTP Basic (Username/Password):** The HTTP Basic authentication module enables username/password authentication against HTTP proxies that implement the RFC 2068 HTTP Basic authentication protocol.

NOTE: *Not all versions of Aventail Connect have encryption enabled.*

- **Configuration Files:** Aventail Connect needs at least one configuration (`.cfg`) file in order to function properly. The configuration file contains all of the authentication and traffic routing instructions that you specify. You can include one or more configuration files in the setup package; however, each configuration file must have a different name. If you include only one configuration file in a setup package, Aventail Connect will automatically use that configuration file. If, however, you include multiple configuration files, Aventail Connect will prompt users to select a configuration file at startup.

  You can include local configuration files, remote configuration files, or a combination of both. Local configuration files are included in the setup package and are installed on users' machines. If you include remote configuration files, pointers to those files are included in the package; the remote configuration files remain in their original location on the network, where they can be shared by multiple users.

  If your setup package does not already contain a configuration file, you can add a configuration file to the package. If your setup package contains one or more configuration files, you can remove or replace any or all of the existing configuration files, or you can leave them, unchanged, in the package. If you are upgrading from an earlier version of Aventail Connect, you may not need a new configuration file.
- **License Files:** Aventail Connect requires a valid license file in order to function properly. If your setup package contains a license file, you can remove or replace the existing license file, or you can leave it, unchanged, in the package. If your setup package does not contain a

license file, you can add one to the packaged Aventail license file, `aventail.alf`.

CAUTION: *Aventail Connect 3.01 and 2.51 use a different license (`.alf`) file format than earlier versions of Aventail Connect (VPN Client or AutoSOCKS) did. If you are upgrading from an earlier version of Aventail Connect (2.42 or earlier), you must include a new Aventail license file.*

- **Extranet (SEE) Hosts Files:** Secure Extranet Explorer (SEE) allows you to browse remote computers using Extranet Neighborhood. SEE requires a hosts file that specifies which Windows domains, WINS servers, and other computers are available in Extranet Neighborhood. The extranet hosts (SEEHosts) file is contained in the setup package. If you install SEE, this file is placed in the target directory. If you do not include a hosts file in the setup package, Aventail Connect will automatically create a hosts file on users' machines the first time they open Extranet Neighborhood. (Available only in Windows 95, Windows 98, and Windows NT 4.0.)

Creating, Loading, and Saving Packages

You can create, load, or save custom setup packages through either the Customizer Editor or the Customizer Wizard.

To create a new package

There are two ways to create a new custom setup package:

- In the **Customizer Editor** window, select **File | New**.

-OR-

- Type the filename of a new package in the first window of the Customizer Wizard and click **Next**.

To load a package

There are two ways to load an existing setup package:

- In the **Customizer Editor** window, select **File | Open**, and then enter the filename of the package you want to load

-OR-

- Type the filename of the package in the first window of the Customizer Wizard and then click **Next**.

When you load a package, Customizer reads the setup control file to determine what information the package contains. Customizer uses this information to populate the **Customizer Editor** window. Customizer also reads the configuration file(s) into memory; configuration files are stored in memory to facilitate adding them to and removing them from a package.

To save changes to a package

There are two ways to save changes to a setup package:

- After making the desired changes to the package, click **Save** (or **Save As**) on the **File** menu in the **Customizer Editor** window

-OR-

- Click **Save Package** in the final window of the Customizer Wizard.

CUSTOMIZER TIPS

The following tips will help you use the Aventail Customizer more efficiently.

- **Keep the package size small:** You can control the size of your custom setup packages by selecting components carefully. To keep the package as small as possible, include only the options that you need, and support only the platforms (e.g., Windows 98, Windows NT 4.0, etc.) that your users work with. You may find that creating two separate, smaller packages is preferable to creating one larger package. For example, you might create one package that supports Windows 98 and Windows NT 4.0 operating systems, and another separate package that supports Windows 3.1 and Windows 95 operating systems.
- **Use descriptive package names:** When naming setup packages, assign descriptive, recognizable names that will help users identify the setup packages.
- **Select components carefully:** If you include the Config Tool in the package, users will be able to view and modify the settings in the Config Tool. Aventail recommends that, in most cases, you do not include the Config Tool in your custom setup package(s). Excluding options such as the Config Tool will eliminate users' ability to modify your settings, and will keep the package size smaller. However, the S5 Ping and Logging Tool utilities are useful diagnostic tools, and Aventail recommends including these options in the setup package whenever possible.
- **Install Aventail Connect 2.51 on Windows 95:** By default, Windows 95 does not support WinSock 2.0, but you can upgrade it to support WinSock 2.0 with a Microsoft patch. (The patch, `w95ws2setup.exe`, is available from Microsoft, at **http://www.microsoft.com/windows/downloads/contents/Updates/W95Sockets2/default.asp**. However, this procedure adds an extra step to the installation and setup process. Unless users need the Multi-Proxy feature, which is available only in Aventail Connect 3.01, Aventail recommends that you install Aventail Connect 2.51 rather than 3.01 on machines running the Windows 95 operating system.
- **Include a hosts file:** If you install Secure Extranet Explorer (SEE) without also installing a corresponding hosts file, SEE will automatically create a hosts file the first time that users open SEE. If you want to control which hosts users can view, Aventail recommends that you include a hosts file in the custom setup package.

- **Include a license file:** Aventail Connect requires a valid license file (`aventail.alf`) to function properly. Aventail Connect 3.01/2.51 uses a different license file than earlier versions of Aventail Connect (VPN Client or AutoSOCKS) did. If you are upgrading from an earlier version of Aventail Connect (2.42 or earlier), you must use the new Aventail license file, `aventail.alf`. Including this license file in the custom setup package is a simple way to install the license file.
- **Test each custom package:** Aventail recommends that you thoroughly test each custom setup package before distribution to users.

CONFIGURING AVENTAIL CONNECT

Create configuration files using the Config Tool or the Configuration wizard. You can launch either during the Aventail Connect installation or any time you want to add, modify, or remove a configuration file.

The steps for creating a new configuration file are:

1. Define the extranet (SOCKS) servers
2. Define the destinations (networks and hosts)
3. Specify redirection rules
4. Enter Local Name Resolution information (optional)
5. Manage authentication modules
6. Enable password protection (optional)

These procedures are described in the text below.

To launch the Config Tool

The Config Tool opens with the **Open Aventail Connect Configuration File** dialog box. After you select a configuration file or enter a new file name, the main window of the Config Tool appears.

1. Select the **Yes, I want to configure Aventail Connect** box in the **Setup Complete** dialog box (during installation).

-OR-

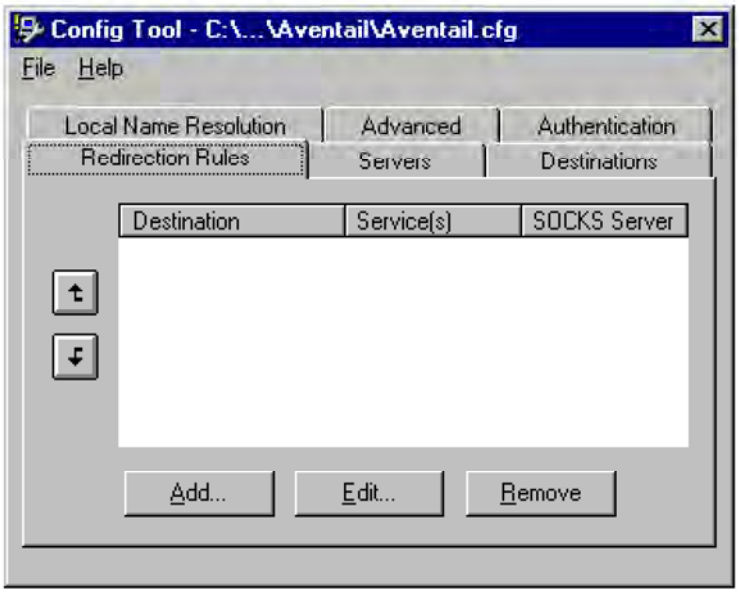
Right-click the **Aventail Connect** icon in the taskbar and click **Config Tool** (Windows 95, Windows 98, or Windows NT 4.0 programs menu option), or double-click the **Config Tool** icon in the Aventail Connect program group (Windows 3.1, Windows for Workgroups 3.11, or Windows NT 3.51).

2. If you are creating a new configuration file, enter a name for the configuration file

-OR-

Select the configuration file you want to open.

This displays the main window of the Config Tool.



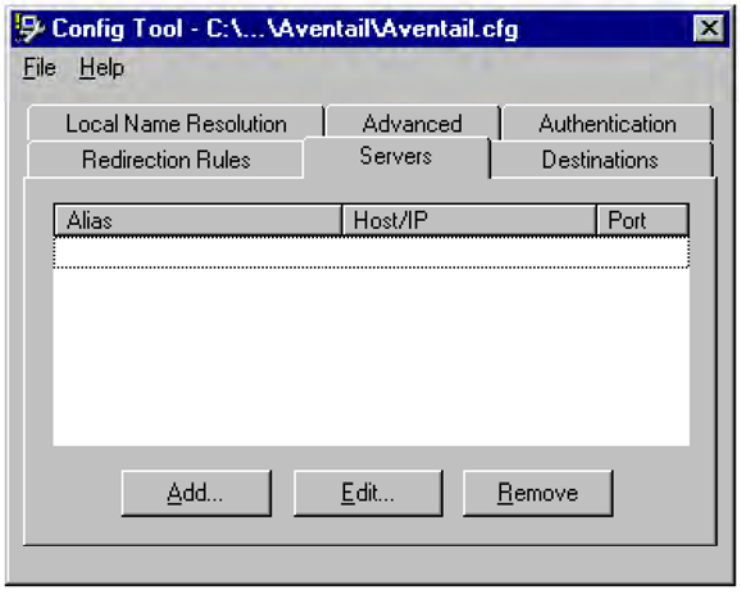
The **Config Tool** window contains six tabs. The properties defined on each tab can be edited at any time.

Tab	Function
Servers	Defines the extranet (SOCKS) server(s).
Destinations	Specifies the network and host addresses that will be routed through the SOCKS server(s).
Redirection Rules	Specifies how network requests are routed to the SOCKS server(s).
Local Name Resolution	(Optional) Specifies hostnames that will be resolved by the local workstation.
Authentication	Enables, disables, and sets properties for the authentication modules.
Advanced	Enables/disables extranet (SOCKS) traffic through successive SOCKS servers, enables/disables the Application Exclusion/ Inclusion List, secures selected applications, and sets credential cache timeouts.

You can change the width of any of the fields on the tabs by positioning the cursor over the dividing line between the fields on the field bar. When the cursor changes to a double-headed arrow, click and drag to resize the field.

DEFINE AN EXTRANET (SOCKS) SERVER

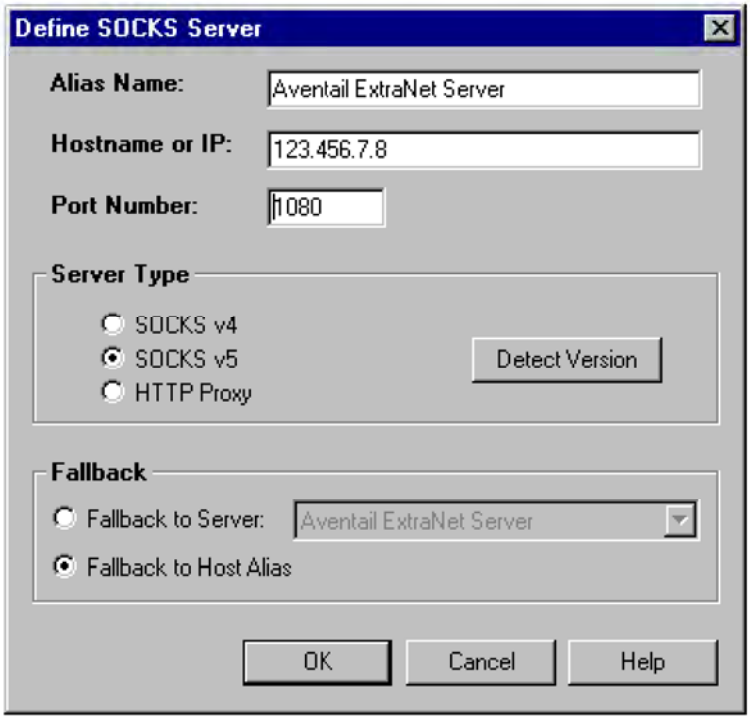
SOCKS servers are defined on the **Servers** tab in the Config Tool.



Field	Definition
Alias	The name you assign to the server.
Host/IP	The hostname or IP address of the server.
Port	The port on which the server is listening.

To add an extranet (SOCKS) server

1. On the **Servers** tab, click **Add...**. The **Define SOCKS Server** dialog box appears.



Field	Definition	
Alias Name	User-friendly alias for extranet (SOCKS) server.	
Hostname or IP	Actual hostname or full numeric IP address for SOCKS server.	
Port Number	SOCKS server port. Default value is 1080.	
Server Type	SOCKS v4	SOCKS Version 4.0.
	SOCKS v5	SOCKS Version 5.0.
	HTTP Proxy	HTTP proxy server.
	Detect Version	Detect SOCKS version number.
Fallback	Fallback to Server:	SOCKS server alias for redundant server.
	Fallback to Host Alias	Use DNS records for redundancy.

2. In the **Alias Name** box, type a user-friendly alias for the extranet (SOCKS) server. Do not leave this box blank.

3. In the **Hostname or IP address box**, type the actual hostname of the SOCKS server or its IP address.
4. In the **Port Number** box, type the extranet server's port number. If you do not enter a value, it defaults to the standard SOCKS port 1080.
5. Under "Server Type," select the version of SOCKS supported by the server. If you are unsure of the version, click **Detect Version**.

NOTE: *Typically you should select **SOCKS v5** unless the server can support only SOCKS v4.*

6. Under "Fallback," directly specify an extranet server for redundancy or use the Host Alias to specify an extranet server.

To edit extranet (SOCKS) server properties

- Select the extranet server you want to edit and click **Edit**.

  The **Define SOCKS server** dialog box appears with the selected server data filled in. Edit any of the information.

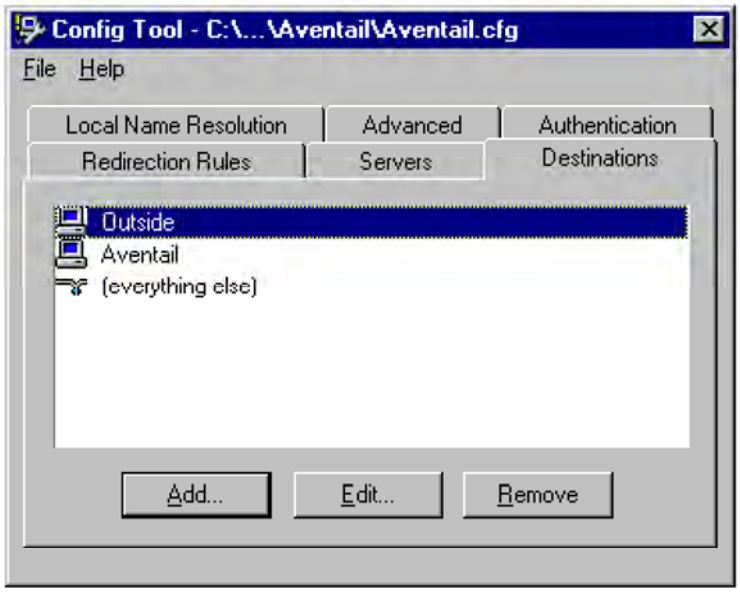
To remove an extranet (SOCKS) server definition

- Select the extranet server you want to remove and click **Remove**.

  The server is deleted from the list. Corresponding redirection rules will also be deleted.

DEFINE A DESTINATION

Destinations are defined on the **Destinations** tab in the Config Tool.



After one or more extranet servers are defined, add destinations to be routed through them.

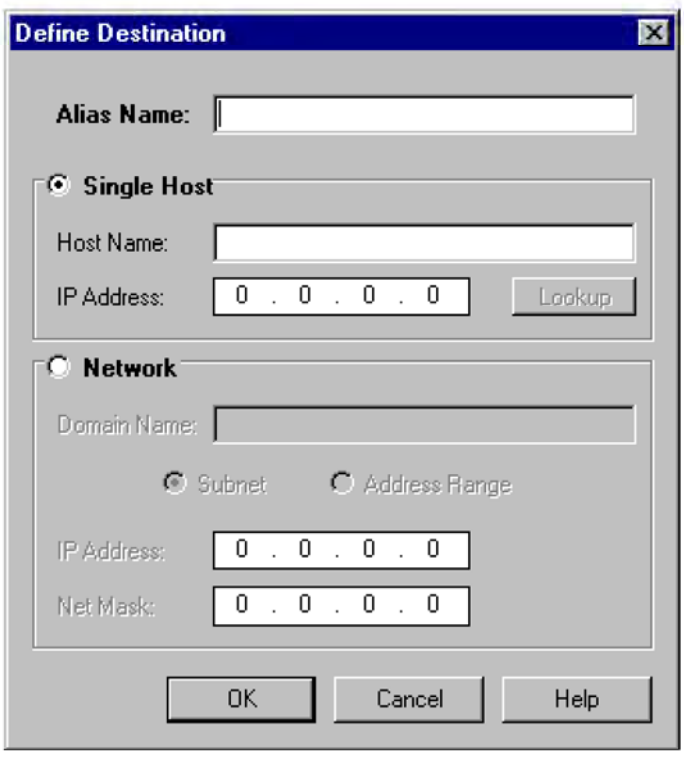
NOTE: *The "(everything else)" destination refers to all network and host addresses not otherwise defined. You cannot delete or modify "(everything else)".*

To add a destination

In the **Define Destination** dialog box, you can define subnets, individual host computers, or IP address ranges, and set up rules about redirecting some or none of the IP traffic to these defined destinations.

1. On the **Destinations** tab, click **Add...**.

The **Define Destination** dialog box appears.



Field	Definition	
Alias Name	User-friendly alias for destination network or host	
Single Host	A specific destination computer	
	Hostname	Actual name of destination network or host
	IP Address	Full numeric IP address
	Lookup	Look up IP address
Network	One or more computers in a network	
	Domain Name	Domain of the network
	Subnet	IP address and netmask address
	Address Range	Beginning and ending IP addresses From Starting IP address To Ending IP address

2. In the **Alias Name** box, type a user-friendly alias for the destination network or host.
3. Select either the **Single Host** or **Network** option:
   - Under "Single host," type the actual name of the host system and/or its full, numeric IP address. If you do not know the host's IP address, click **Lookup** to search for it.

-OR-

   - Under "Network," type the domain of the network and then select either **Address Range** or **Subnet**.

Use	To
Address Range	Enter a starting and ending IP address. All addresses between the two will be included as part of the destination. For example, a starting IP address of 192.1.1.0 and an ending IP address of 192.1.1.255 would include all hosts of the 192.1.1.x subnet.
Subnet	Enter an IP address and a netmask address. This is another way to specify a group of destinations. For example, an IP address of 192.1.1.0 and a net mask of 255.255.255.0 defines the same address range as shown above.

To edit a destination

- Select the destination you want to edit and click **Edit...**.

  The **Define Destination** dialog box appears with the selected destination data filled in. Edit the data as necessary.

To remove a destination

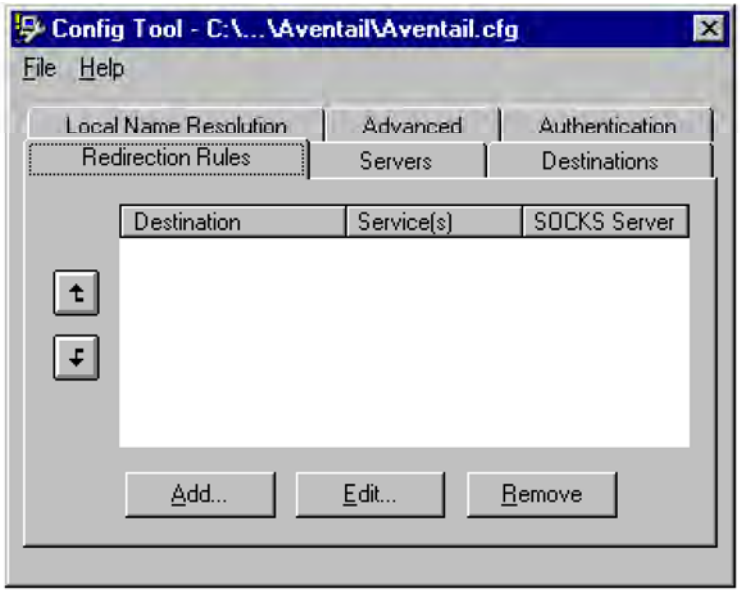
- Select the destination you want to remove and click **Remove**.

  The destination is deleted from the list. The corresponding redirection rules will also be deleted.

ENTER REDIRECTION RULES

Once servers and destinations are defined, you can specify how you want Aventail Connect to redirect (or deny) access to various hosts and services such as e-mail, FTP, and HTTP.

Redirection rules are specified on the **Redirection Rules** tab in the Config Tool.



.

Field	Definition
Destination	Destinations defined on the **Destinations** tab
Service	Type of Internet traffic
Proxy Redirection	Specify how to redirect traffic

You can change the width of any of the three fields by moving the cursor to the dividing line between the fields on the field bar. When the cursor changes to a double-headed arrow, click and drag to resize the field.

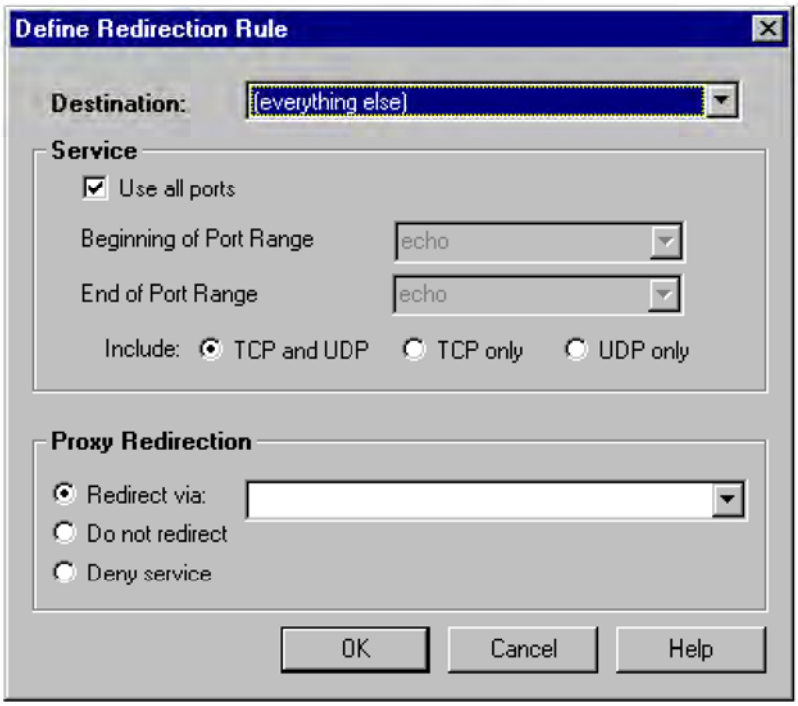
To add a redirection rule

As you add destinations, use the arrow buttons to prioritize them. List the most specific rules first and the general rules last.

NOTE: *Aventail Connect scans the list from the top down and uses the first matching rule it finds, so it is important to list the most specific rules first.*

1. On the **Redirection Rules** tab, click **Add**.

The **Define Redirection Rule** dialog box appears.



Field	Definition	
Destination	Host or server destination for message traffic.	
Service	Type of Internet traffic	
	Use all ports	Apply the defined rule to all ports.
	Beginning of port range	Apply the defined rule to this range of ports.
	End of port range	
	TCP and UDP	Apply the defined rule to both TCP and UDP traffic.
	TCP only	Apply the defined rule to TCP traffic only.
	UDP only	Apply the defined rule to UDP traffic only.
Proxy Redirection	Specify how to redirect traffic.	
	Redirect via	Redirect all traffic through the extranet server selected from the list.
	Do not redirect	Route traffic directly to the specified destination without being redirected through SOCKS.
	Deny service	Deny access to the specified destination. The network connection is blocked locally instead of at the server level.

2. Select a destination from the **Destination** list.
3. Under "Service," select the **Use all ports** box to apply the rule to all services. Otherwise, select a range of ports. To select a single port, enter that port number in both the **Beginning of port range** and **End of port range** boxes.
4. Under "Proxy Redirection," select one of three redirection options.

CAUTION: *If you select **Deny Service** and the user has edit control of the configuration file, the option can be circumvented by quitting Aventail Connect or by changing the option in the dialog box.*

To edit a redirection rule

- Select the redirection rule you want to edit and click **Edit...**.

  The **Define Redirection Rule** dialog box appears with the selected data filled in. Edit any of the information.

To remove a redirection rule

- Select the redirection rule you want to remove and click **Remove**.

  The redirection rule is deleted from the dialog box.

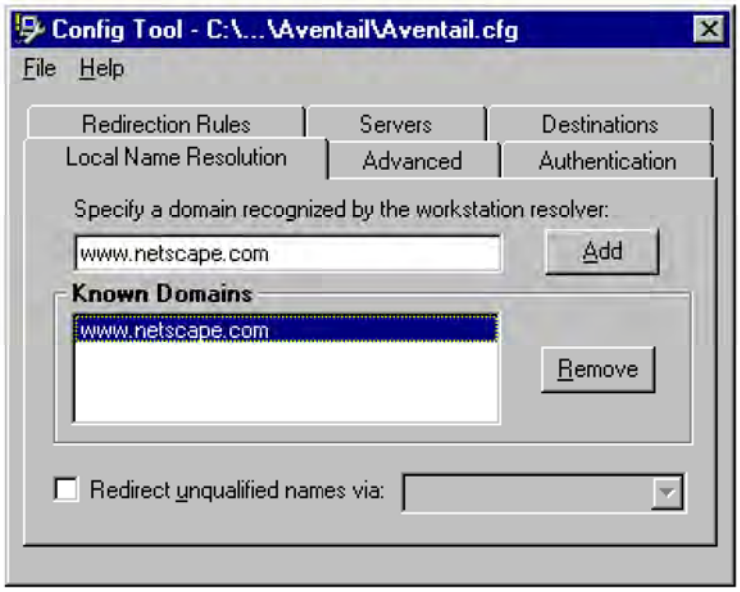
DEFINE LOCAL NAME RESOLUTION

Local Name Resolution instructs Aventail Connect to resolve hostnames locally without needing to venture on to the Internet. This optional feature offers you another level of control over how Aventail Connect performs name resolution.

The local workstation resolver is the name resolution component of the local TCP/IP stack. This feature acts as a shortcut; hostnames matching the strings defined in the **Local Name Resolution** dialog box are passed to the local resolver for name resolution instead of being proxied through the SOCKS v5 server.

For example, if **aventail.com** is added to the "Defined Strings" list, then a workstation attempting to connect to **www.aventail.com** would perform hostname resolution using the local TCP/IP stack.

Local Name Resolution is specified on the **Local Name Resolution** tab in the Config Tool.



Field	Definition
Specify a domain recognized by the workstation resolver	New domain name
Known Domains	List of domain names that can be resolved locally
Redirect unqualified names via	Pass through unqualified hostnames to the local resolver

To add a local domain name

- On the **Local Name Resolution** tab, type the new name in the **Specify a domain** box and click **Add...**.

  The new name is moved into the **Known Domains** box. It is now active.

To remove a local domain name

- Select the domain name you want to remove from the **Known Domains** box and click **Remove**.

  The domain name is removed from the list.

MANAGE AUTHENTICATION MODULES

SOCKS v5 servers often require user authentication before allowing access. Aventail Connect authentication modules display dialog boxes that prompt users to enter username and password information as well as other authentication credentials.

NOTE: *Not all versions of Aventail Connect have encryption enabled.*

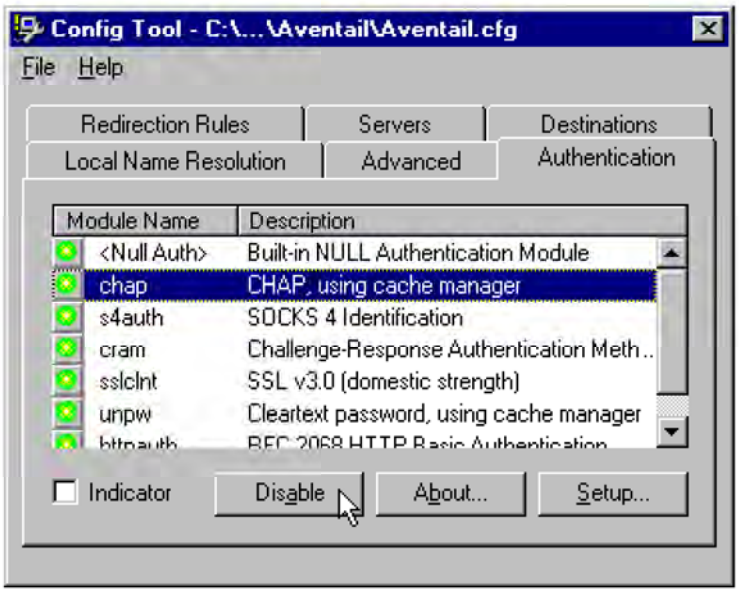
The current Aventail Connect authentication modules are SOCKS v4 Identification, Username/Password, Challenge Handshake Authentication Protocol (CHAP), Challenge Response Authentication Method (CRAM), Secure Sockets Layer (SSL), and HTTP Basic (username/password). Each of these authentication modules supports an Aventail Connect feature known as credential caching. Credential caching retains your authentication credentials once the extranet server has accepted them. Using credential caching, you can enter your credentials for an extranet server once per Aventail Connect session, rather than once for each individual connection (a tedious task for applications such as WWW browsers).

Aventail Connect can cache authentication credentials in memory, based on the option you select in the **Authentication** dialog box. Memory caching stores the credentials for the current session only. When you restart Aventail Connect or

Windows, the memory cache is flushed and you must reenter your credentials as prompted.

SEE ALSO: *For additional information on credential caching, see "Credential Cache Timeouts" in the "Advanced Tab Options" section of this Administrator's Guide.*

Authentication modules are managed and configured through the **Authentication** tab in the Config Tool.



Field	Definition
Module Name	The name of the authentication module on disk. `<Null Auth>` indicates that no authentication module will be used.
Description	The description of the authentication method.
Indicator	Check this option to display network traffic passing through a selected authentication/encryption module. See the example below (for Windows 95, Windows 98, and Windows NT 4.0). 

Each authentication module includes its own module-specific configuration. To view or edit a module's configuration, select the module from the list on the

Authentication tab and then click **Setup**. An options dialog box for the specific module will appear.

Enable and disable authentication modules with the **Disable/Enable** button. By default, the modules are all enabled. The green button next to the module name indicates an active module. This is the default state of all the modules. The green button changes to red when you disable the module.

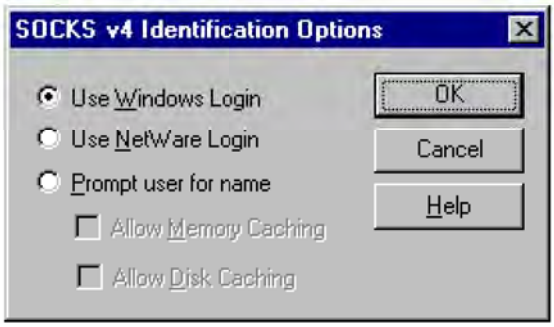
To configure the SOCKS 4 Identification module

Aventail Connect includes backward compatibility for the SOCKS 4 protocol. SOCKS 4 does not support password authentication, so only your username is sent unencrypted to the extranet (SOCKS) server along with your connection request.

Your username is determined by entries in the **SOCKS 4 Identification Module Configuration** dialog box.

1. On the **Authentication** tab in the Config Tool, click **s4auth** (SOCKS v4 Identification) and click **Setup**.

   The **SOCKS 4 Identification Options** dialog box appears.



Field	Description	
Use Windows Login	Identify users by their Windows Login names.	
Use NetWare Login	Identify users by their Novell NetWare Login names.	
Prompt user for name	Identify users by the names they enter for this specific purpose.	
	Allow Memory Caching	Stores credentials in memory for this session only. Cache is flushed upon restart; credentials must be reentered as prompted.
	Allow Disk Caching	This option is currently unavailable. (Stores credentials on disk for future sessions.)

2. When you select the **Prompt user for name** option, you must also select the desired caching option. (Currently only Memory Caching is available.)
3. After making appropriate selections, click **OK**.

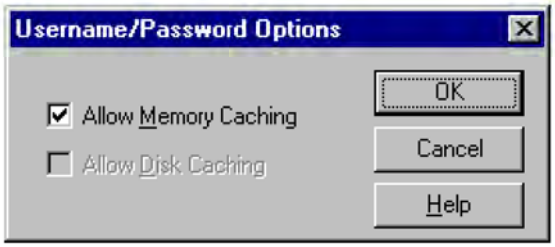
   The dialog box closes and the Config Tool reappears.

To configure the Username/Password authentication module

Aventail Connect supports the RFC 1928 (Internet standards document) username and password authentication protocol. This authentication method sends your username and password *in cleartext* across the network to the destination server. The **Username/Password authentication module** dialog box contains only credential caching options.

1. On the **Authentication** tab in the Config Tool, select **unpw** and click **Setup**.

   The **Username/Password Options** dialog box appears.



Field	Description
Allow memory caching	Stores credentials in memory for this session only. Cache is flushed upon restart; credentials must be reentered as prompted.
Allow Disk Caching	This option is currently unavailable. (Stores encrypted credentials on disk for future sessions.)

2. The selection defaults to **Allow Memory Caching**. Click **OK**.

   The dialog box closes and the Config Tool reappears.

To configure the CHAP authentication module

Aventail Connect supports the Challenge Handshake Authentication Protocol (CHAP). This authentication method sends your username and password *encrypted* across the network to the destination server. The **CHAP authentication module** dialog box contains only credential caching options.

1. On the **Authentication** tab in the Config Tool, select **chap** and click **Setup**.

The **CHAP Options** dialog box appears.

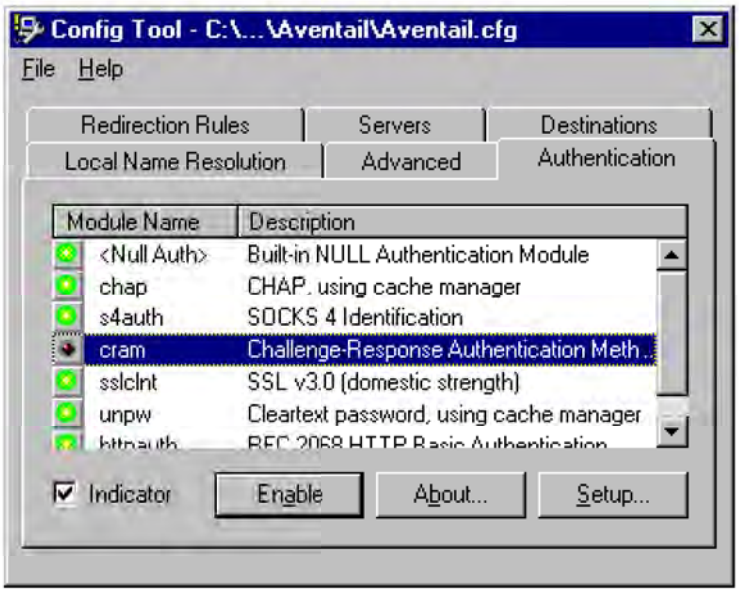
Field	Description
Allow memory caching	Stores credentials in memory for this session only. Cache is flushed upon restart; credentials must be reentered as prompted.
Allow disk caching	This option is currently unavailable. (Stores encrypted credentials on disk for future sessions.)

2. The selection defaults to **Allow Memory Caching**. Click **OK**.

   The dialog box closes and the Config Tool reappears.

To configure the CRAM authentication module

Aventail Connect supports the Challenge Response Authentication Method (CRAM). This authentication method sends your username and passcode as cleartext between extranet (SOCKS) servers, but *encrypted* between servers that support CRAM. Typically, CRAM subauthenticates within SSL, which provides both encryption and credential caching options.



You do not need to configure the CRAM authentication module. You can enable/disable it, by clicking on the **Disable/Enable** button. The button at the left of the module name will change from green to red, accordingly.

To configure the SSL security module

Aventail Connect supports Secure Sockets Layer (SSL) 3.0, a session-layer protocol for securing connections in a general, protocol-independent fashion.

NOTE: *Currently, SSL is a TCP-only enhancement. When using SSL with User Datagram Protocol (UDP) applications, bulk data is passed without encryption.*

Normally SSL servers are required to have an RSA key pair and a certificate. Aventail uses an RSA algorithm to create a cryptographic system: a private key (which, as the name suggests, is kept absolutely private and never shared) and a public key (which is widely published).

NOTE: *In versions of Aventail Connect that do not include encryption, SSL is not available.*

However, as the client, you normally must then establish some kind of relationship between your RSA public key and the identity of the server, so that somebody else cannot create their own RSA key information and use it to impersonate your server. *Certificates* establish this relationship. A certificate is essentially an electronic "statement" that verifies that a certain RSA public key is associated with a particular name.

Certificates are issued by a Certification Authority (CA), and are linked together to form a construct called a certificate *chain of authorities*, each one having a previous entity vouching for its identity. In practice, chains generally include two certificates: one confirming the identity of the server, and the other—a "root" certificate—containing the identity and public key of the CA.

Certificates contain special integrity checks and electronic signatures that verify that the certificate is genuine, was issued by a certification authority, and was not tampered with. Anybody can issue a certificate that says anything; the client must know who issued the certificate, and have some trust relationship in order to believe that it is in fact true. The client has a list of trusted CAs. A set of certificate chains can be structured as a tree, with new certificates stemming from old ones. A base CA is sometimes called the “root” or "trusted root" of this tree.

It is becoming common practice for both clients and servers to exchange certificate information. However, in Aventail Connect the client-side of this exchange is transparent. The client only needs to deal with the information from the server certificate and this is done through the SSL module.

The **SSL module** dialog box contains an initial set of options regarding the viewing of certificates.

1. On the **Authentication** tab in the Config Tool, select **sslclnt** (SSL 3.0) and click **Setup**.

   The **SSL Options** dialog box appears.

Field		Description
Upon Successful Connection		The certificate is valid.
	View when the server certificate is new.	Upon successful connection, display the server certificate if it has not been displayed during the current session.
	Do not show me the certificate.	Never display a valid server certificate.
If a server certificate is suspect		The certificate may not be valid.
	Always show me suspect certificates.	Each time Aventail Connect suspects a certificate may not be valid, show the certificate.
	Show me the same suspect certificate once.	Once a suspect certificate has been accepted by the user, do not display it again.
	Show me the certificate, but reject the connection.	Reject the connection, but display the suspect certificate.

2. Select an action that Aventail Connect must take once it accepts the validity of the server certificate. (Under normal circumstances, the server will provide Aventail Connect with a certificate to match one of Aventail Connect's trusted roots, if any exist):
   - **View when the server certificate is new:** Aventail Connect displays the certificate the first time it is seen. The certificate will not appear on subsequent connections to the same extranet server.
   - **Do not show me the server certificate:** Aventail Connect will never display a valid certificate.
3. Select an action that Aventail Connect must take if it receives a server certificate that is suspect: