

UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE PATENT TRIAL AND APPEAL BOARD

---

Apple Inc.  
Petitioner,

v.

VirnetX, Inc. and Science Application International Corporation,  
Patent Owner

Patent No. 7,490,151  
Issued: Feb. 10, 2009  
Filed: Sep. 30, 2002

Inventors: Edmund C. Munger, *et al.*

Title: ESTABLISHMENT OF A SECURE COMMUNICATION LINK BASED  
ON A DOMAIN NAME SERVICE (DNS) REQUEST

*Inter Partes* Review Nos. IPR2013-00354

---

**Declaration of Chris Hopen Regarding Prior Art  
and U.S. Patent No. 7,490,151**

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 *inter partes* review proceeding.

Dated: 06/15/2013

Signed: \_\_\_\_\_

## **I. INTRODUCTION**

### **A. Engagement**

1. I have been retained by counsel for Apple Inc. to provide testimony in the above-captioned proceeding regarding certain facts of which I am aware, and to offer my opinions regarding certain issues.

2. In particular, I have been asked to provide my recollections on the distribution and availability of certain documents relating to a number of Aventail products, and the product manuals and written materials distributed with those products. I also was asked to provide my opinions regarding certain statements about how these products worked that were made in district court and Patent Office proceedings.

### **B. Background And Qualifications**

3. My Curriculum Vitae is submitted as Exhibit 1057.

4. I am a citizen of the United States, and reside at 19805 15th Avenue NW, Shoreline, Washington.

5. I received a B.S. in Computer Science from Western Washington University in 1988.

6. I am currently the President of TappIn, Inc., a wholly owned subsidiary of GlobalSCAPE, based in Seattle, Washington. TappIn is the successor entity of HomePipe, a company I founded in 2009. TappIn was acquired by GlobalSCAPE in 2011.

7. Before founding TappIn, I was affiliated with Aventail, Inc. until that company was acquired by SonicWall, Inc. in 2007.

8. I helped co-found Aventail in 1996, and served as its Chief Technical Officer and Vice-President of Engineering from 1996 to 2007.

9. Prior to co-founding Aventail, I served as Director of Network Technology for CompuServe's Internet Division, where I was a key contributor to the development of CompuServe's dial-up internet products and designed and managed the development of the dial-up protocol stacks of SPRY's Internet In a Box, AIR Series, and Internet Office application suites.

10. I am a named inventor on several U.S. patents, including patents related to classifying an operating environment of a remote computer, provisioning remote computers for accessing resources, systems and techniques for controlling requests for resources from remote computers, distributed cache services, controlling access to a set of resources on a network, and rule-based routing to resources through a network.

## **II. Public Availability of Aventail Products**

11. While I was affiliated with Aventail, I was involved in the design, development and distribution of all of Aventail's products.

12. I have personal knowledge of development of the products, and with their distribution to Aventail customers.



13. I am also very familiar with the technical features of the products, as I was involved in creating them.

**A. Aventail AutoSOCKS**

14. In 1997, Aventail released a set of SOCKS v5 compliant VPN software products including AutoSOCKS v2.1 (“AutoSOCKS”), MobileVPN and PartnerVPN.

15. AutoSOCKS was a client-based software product that ran on users’ computers, while Mobile VPN and Partner VPN were server-based products.

16. When paired with the Aventail MobileVPN or PartnerVPN server products, Aventail AutoSOCKS would automatically establish a VPN to give the remote user access to secured network resources on a private network.

17. The AutoSOCKS client and the server would automatically authenticate the remote user and encrypt all communications between the remote user and the destination network.

18. Version 2.1 of the AutoSOCKS product was publicly distributed in the summer of 1997.

19. Aventail issued press releases announcing it was selling and distributing these products. Exhibit 1058 is a copy of a May 2, 1997 Aventail press release announcing the AutoSOCKS v2.1, MobileVPN, and PartnerVPN

products. Exhibit 1059 is a copy of a June 23, 1997 article in InfoWorld reviewing the AutoSOCKS v2.1 and MobileVPN v2.0 products.

20. Aventail included printed manuals with all of the software packages that it distributed.

21. Exhibit 1021 is a copy of the Aventail AutoSOCKS v2.1 Administrator's Guide that was distributed with the AutoSOCKS v2.1 software. This document was distributed without any confidentiality restrictions.

22. I estimate that thousands of copies of the Aventail AutoSOCKS v2.1 software that included the AutoSOCKS v2.1 Administrator's Guide were distributed to customers during 1997 and 1998.

**B. Aventail Extranet Center (AEC) v3.0**

23. In the fall of 1998, Aventail announced a product called the Aventail Extranet Center ("AEC").

24. Aventail issued a press release announcing that this product was available in the fall of 1998. Exhibit 1060 is a copy of an October 12, 1998 Aventail press release announcing the Aventail Extranet Center 3.0 product. Exhibit 1061 is a copy of an October 19, 1998 Network World publication announcing the Aventail Extranet Center 3.0 product.

25. The AEC product had three components: (i) the Aventail Extranet Server (which resided and ran on a server computer), (ii) the Aventail Management

Server and Config Tool (which was used to configure server and client installations), and (iii) the Aventail Connect client software (which resided and ran on client computers).

26. The initial release version of the AEC product was version 3.0. The AEC v3.0 product included version 3.01/2.51 of Aventail Connect and version 3.0 of the Aventail Extranet Server.

27. The AEC product was distributed with printed manuals, including the Aventail Connect v3.01/2.51 Administrator's Guide and the Aventail Extranet Server v3.0 Administrator's Guide.

28. Exhibit 1007 is a copy of the Aventail manuals that were distributed with the AEC v3.0 product.

29. The Aventail manuals were an interrelated set of documentation for the Aventail AEC v3.0 products. They were designed to be used together to help administrators configure and use various components that made up the AEC product. For example, the Aventail Connect and Extranet Administrator Guides refer repeatedly to each other when describing how to configure and use the components of the AEC product. *See, e.g.*, Ex. 1007 (Aventail) at 5, 10-12, 14, 127 ("Instructions covering advanced configuration options, public certificates, and troubleshooting may be found in the 'Aventail Connect Administrator's Guide' and in the online Help.")

30. The Aventail manuals were distributed without any confidentiality restrictions. Specifically, Aventail customers were not required to accept any obligations limiting their ability to use or disseminate the information contained in or associated with the AEC v3.0 product, or the manuals that accompanied the product.

31. The AEC v3.0 product, along with the Aventail manuals, was being sold and distributed to hundreds of customers before January of 1999.

32. I personally recall that the AEC v3.0 product was distributed publically at least as early as October 1998.

33. I also personally distributed copies of the AEC v3.0 product to customers before January of 1999.

34. In particular, I personally distributed copies of installation CDs containing version 3.01/2.51 of the Aventail Connect Client and version 3.0 of the Aventail Extranet Server, along with the Aventail manuals, to customers and potential customers at least as early as October 1998.

35. In addition, when the product began shipping to customers in the fall of 1998, Aventail made the Aventail manuals (including the Administrator's Guides for Aventail Connect and Extranet Center, and the Quick Start Guide) available via download from the Aventail website, [www.aventail.com](http://www.aventail.com).

36. I estimate that Aventail distributed thousands of copies of the AEC v3.0 product (including the Aventail) during the first six months of 1999 alone.

### **III. Technical Features and Capabilities of Aventail Products**

37. All of my comments in this section will concern the Aventail v3.0 product described in Ex. 1007 (Aventail).

38. Like the earlier Aventail VPN solution (AutoSOCKS), Aventail Connect would, when paired with the Extranet Server, automatically establish a VPN between a remote user and a private network to give the remote user access to secured network resources on a private network. Ex. 1007 (Aventail) at 15-16.

39. The Aventail Connect client and the Extranet Server could be configured to automatically authenticate the remote user and encrypt all communications with the remote user. Ex. 1007 (Aventail) at 16, 76-77.

40. The Aventail Connect client and Extranet Server products could be configured to use different authentication techniques, including SOCKS v4 Identification, Username/Password, Challenge Handshake Authentication Protocol (CHAP), Challenge Response Authentication Method (CRAM), Secure Sockets Layer (SSL) authentication, and HTTP Basic (username/password). Ex. 1007 (Aventail) at 46-63.

41. The Aventail Connect client and Extranet Server products could be configured to use different encryption algorithms and techniques, including SSL

encryption and Diffie-Hellman Anonymous encryption, and RC4 and DES ciphers. Ex. 1007 (Aventail) at 55.

42. The AEC v3.0 product operates on both “inbound” and “outbound” connections. Whether a connection was “inbound” or “outbound” is simply a matter of perspective – an “outbound” request for access to a secure target computer intercepted by Aventail Connect on the client computer would be an “inbound” connection from the Extranet Server and the target computer perspective. I also note that the Aventail Connect user manual points out that the communications with a client computer were bi-directional – the client not only sends “outbound” requests but can respond to “inbound” requests. *See, e.g.*, Ex. 1007 (Aventail) at 11 (“You can use Aventail Connect as a simple proxy client for managed outbound access, and for secure inbound access.”)

43. I understand that VirnetX, in previous proceedings, has asserted that its alleged invention is distinguishable from the Aventail system embodied in the AEC v3.0 product for a variety of technical reasons. *See generally* Ex. 1051 (Keromytis Decl.) and Ex. 1052 (Nieh Decl).

44. For example, VirnetX has incorrectly claimed that the AEC v.30 product did not establish a virtual private network. Ex. 1051 (Niehs Decl) at 4. VirnetX also incorrectly described how the various components of the AEC v3.0

product could be configured, and how they work together. Ex. 1052 (Keromytis Decl) at 7-10.

45. The Aventail AEC v3.0 product would transparently create a virtual private network (VPN) between a client computer and a private network. First, the Aventail Connect client running on the client computer would intercept and evaluate all TCP/IP application calls on the client computer (e.g., DNS requests made through a web browser). Either the client computer, or the server if requests were being proxied, would determine if the request specified a secure destination (e.g., a host on the private network). If it did, the client would be authenticated and communications between the client computer and the private network would be automatically encrypted. Ex. 1007 (Aventail) at 11-16.

46. I understand that VirnetX has claimed the AEC v3.0 product would not create a VPN. *See* Ex. 1051 (Nieh Decl) at ¶ 19. I disagree.

47. The Aventail Extranet Server worked in conjunction with the Aventail Connect Client to establish an encrypted communication channel over the Internet that would allow a client computer to communicate privately over the Internet with a secure destination on the private network. Ex. 1007 (Aventail) at 13. These are VPNs. The Aventail manuals even refer to the Aventail ExtraNet Server as being the “Aventail VPN Server” in the secure extranet example in the Aventail manuals. *See, e.g.,* Ex. 1007 (Aventail) at 76.

48. I understand that VirnetX has claimed that computers connected with AEC v3.0 cannot communicate directly with each other as though they were on the same network. *See* Ex. 1051 at ¶¶ 20-22, 26-27 (Nieh Decl). I disagree. In fact, if this were true, we would not have been able to market a viable product. The whole point of the Aventail technology and product line was to enable a remote user to communicate through a VPN with other computers on a private, secure network.

49. Computer applications connected by Aventail Connect and ExtraNet Center could communicate directly with each other as though they were on the same network. For example, the AEC v3.0 product included a feature called the “Extranet Neighborhood.” This feature included a Windows Explorer-type interface called “Secure Extranet Explorer” that allowed remote users connected using Aventail Connect to browse and access specific hosts and other network resources on the corporate network. *See* Ex. 1007 (Aventail) at 30-31, 94-95.

50. The Secure Extranet Explorer would allow a remote user connected to the network using Aventail Connect to browse, copy, move, and delete files on target host computers. *See* Ex. 1007 (Aventail) at 94 (“Extranet Neighborhood offers Aventail Connect users a secure alternative to traditional file-browsing methods. Users can securely access computers from the desktop through Extranet Neighborhood [], or through Windows Explorer.”) The Secure Extranet Explorer thus allowed the remote client computer to access specific network resources on



the private computer as though the target host computer and the remote computer were on the same internal network. Ex. 1007 (Aventail) at 94-95.

51. I also understand that VirnetX has claimed that the fundamental operation of the AEC v3.0 product is incompatible with users transmitting data that is sensitive to network information. *See* Ex. 1051 (Nieh Decl) at ¶¶ 23-25. Again, that is incorrect.

52. The central purpose of the AEC v3.0 product was to allow remote users to access secure (“sensitive”) network resources. The software was designed to give those remote users the ability to connect to and share information as if they were physically connected to the private (e.g., corporate) network. I cannot understand how anyone familiar with the purpose or operation of the AEC v3.0 product cannot appreciate this point, given that this was its central purpose.

53. We designed the AEC v3.0 product to make the entire process of gaining access to private network resources as simple, transparent and automatic as possible. The Aventail v3.0 manuals explain this clearly. Specifically, they show that a user who wants access to a secure network resource need only specify the particular host on the private network, either by entering the hostname or IP address of the target destination in a web browser or by selecting the host directly using the Secure Extranet Explorer. The Aventail Connect client on the user’s computer would then transparently intercept that connection request and proxy it to

the Aventail Extranet Server, which would authenticate the user and automatically establish the VPN between it and the user's computer. Ex. 1007 (Aventail) at 11-16, 46-55, 77, 80-81, 83, 89, 99, 108, 129-131, 141-142, 144-145, 154-158, 160-167. The whole point of this was to automatically and transparently create a VPN between the remote user and the private network that would allow the remote user to securely access the private network.

54. I understand that VirnetX has claimed that the use of false DNS responses in Aventail prevents the correct transfer of data in the Aventail scheme. *See* Ex. 1051 (Nieh Decl) at ¶ 25. I also understand that VirnetX has asserted that there is no connection between the redirection rules and use of false DNS entries and whether the target is secure or not, and that the VPN is not initiated in response to a determination the DNS request was seeking access to secure resources on a private network. Ex. 1052 (Keromytis Decl) at ¶ 24-27. Each point is incorrect.

55. There are three basic steps used by all WinSock applications to establish a connection using a hostname and domain name, namely: (1) a DNS lookup to identify the IP address associated with the hostname and domain name, (2) a request to establish a connection to the remote host, and (3) the transmission and receipt of data through that connection. Ex. 1007 (Aventail) at 12. The Aventail Connect functionality was nested within these steps. Ex. 1007 (Aventail) at 15-16.

56. As the Aventail manuals explain, each connection request would be intercepted by the Aventail Connect client on the client computer. Ex. 1007 (Aventail) at 11, 13-14, 15-16. If the request contained a domain name, the Aventail Connect client would handle evaluation of that domain name in one of a three ways. First, if the client were configured with local name resolution rules, and the domain name matched a value on one of those rules, the domain name would be locally resolved. Second, if the client were configured to proxy all non-locally resolved domain names to a proxy server, the request would be proxied to the designated Extranet Server for name resolution and other handling. Finally, if connection requests were not being proxied, and the domain name did not match a local name resolution rule, the Aventail Connect client on the client computer evaluated only the domain part of the fully qualified hostname to see if it should be securely routed through the Extranet Server. Ex. 1007 (Aventail) at 15-16.

57. In the latter two scenarios, the Aventail Connect client would insert a validly formed yet unused and special IP address into the DNS response to trigger redirection and encryption of that communication through the Extranet Server. Ex. 1007 (Aventail) at 15-16. The Aventail Connect client did not attempt to actually resolve these false domain name entries, and did not treat them as actual domain names specifying destinations. Instead, the flag simply informed the Aventail

Connect client that the request had to be proxied to the Aventail Extranet Server associated with the domain name. Ex. 1007 (Aventail) at 16.

58. In a second step, the client computer would open a connection to the designated proxy server (i.e., the designated Extranet Server), and the client would then be authenticated. Ex. 1007 (Aventail) at 15-16. If authentication were successful, the client and the ExtraNet Server would then transmit the proxied connection request (i.e., the destination of the request) and then data between the client and that host on the private network. Ex. 1007 (Aventail) at 16. All of this data would be automatically encrypted and decrypted by the Aventail Connect client. Ex. 1007 (Aventail) at 16.

59. The use of a false DNS entry by Aventail, thus, did not prevent correct data transfer at any point in this process. Again, it was simply a technique to flag a connection request requiring proxying to the Extranet Server for handling. Ex. 1007 (Aventail) at 15-16. Indeed, the Aventail AEC v3.0 product would not have been a commercial success if it could not perform this basic function of securely routing traffic to and from the remote client to the secure network.

60. I understand that VirnetX has said that the Aventail systems do not return an IP address to the client computer if the DNS request specifies a non-secure website. *See* Ex. 1051 (Nieh Decl) at ¶¶ 30-33. This is incorrect.

61. The way the Aventail Connect client worked shows that it would return an IP address to a requesting application if a domain name in a connection request specified a destination other than one on the remote private network.

62. First, the Aventail Connect client could be configured to resolve domain names using “local name resolution” rules. Domain names matching a local name resolution rule were addresses that could be locally resolved (e.g., local names on the network or by a DNS server on the local network). Names matching a local name resolution rule would not specify a host on a remote private network because these requests would be handled locally, and would not be proxied to the remote network. Also, if local name resolution rules had been created on the Aventail Connect client, a domain name in a request would be compared to these rules first, before the request would be evaluated against a redirection rule or before it would be proxied to the Extranet Server. Ex. 1007 (Aventail) at 15-16.

63. Second, when a user running Aventail Connect made a DNS request that did not match either a local name resolution rule or a redirection rule requiring a VPN, Aventail Connect would pass the request through to the client computer’s native TCP/IP and DNS subsystems, which would then return an IP address for the domain name to the requesting application. See Ex. 1007 (Aventail) at 15-16. This was a well-known “pass-through” technique for resolving unsecured DNS requests at the time we developed the AEC v3.0 product.

64. Third, Aventail Connect could be configured to proxy **all** DNS requests that did not match a local domain name rule to a different computer (the Aventail Extranet Server) for resolution. In this configuration, the content of the connection request was not checked by Aventail Connect on the client computer. Instead, the request was simply proxied to the Extranet Server, which would then evaluate the request. *See* Ex. 1007 (Aventail) at 16 (“The special false response 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.”) If the request contained a domain name for a public website, the Extranet Server would resolve the domain name and return the IP address to the calling application via the Aventail Connect client. All of this was transparent to the calling application that made the connection request. *See* Ex. 1007 (Aventail) at 16 (“From the application’s point of view, the entire SOCKS negotiation, including the authentication negotiation, is merely the TCP handshaking.”)

65. In any of these configurations, the IP address of a “non-secure” destination specified by a domain name will be returned to the client computer. This was important to the design of the product – it helped make the operation of the client “transparent.” Ex. 1007 (Aventail) at 15 (“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.”); Ex. 1007 (Aventail) at 77 (“Second, not all traffic passes through to the Aventail ExtraNet Server. Only traffic destined for the internal network is authenticated and encrypted; all other traffic passes through Aventail Connect unchanged. For instance, browsing the Internet from the mobile user workstation occurs as if Aventail Connect is not even running in the background.”)

66. I understand that VirnetX has claimed that the Aventail systems do not function as a DNS server, and do not return any error message if a DNS request is not successfully resolved. *See* Ex. 1052 (Keromytis Decl.) at ¶ 29. The first point seems irrelevant because none of the claims of the ‘151 patent refer to a DNS server – they speak about “DNS proxy modules” and evaluating “DNS requests.” There is no question the Aventail Connect and ExtraNet Server function as DNS proxy servers – they each handle intercepted DNS requests and evaluate those requests.

67. Aventail Connect and Aventail ExtraNet Server products were inherently DNS-based schemes. And, if a domain name in a request could not be resolved, an error would be returned to the calling application. This is a consequence of how the DNS resolution process works.

68. I note that it was possible to configure Aventail Connect to proxy all requests – including both secure and non-secure destinations – to the Aventail

ExtraNet Server for handling. This would be done, for example, to prevent a user while on a VPN from directly accessing a public website. In this configuration, if the proxied request contained a domain name of a public website (e.g., “apple.com”), the Aventail ExtraNet Server would resolve the name and return the IP address.

69. If the name could not be resolved, an error would be returned. So, for example, if a user failed authentication, an error, typically “host unknown,” would be returned to the calling application. Similarly, if accessing the public website would violate a policy (e.g., visiting a competitor’s website while on the VPN), the Aventail ExtraNet Server would refuse to resolve the domain name, and the “host unknown” error again would be returned to the calling application.

70. I understand that VirnetX has claimed that the IP address-hopping schemes employed by the AEC v3.0 product do not “contribute in any meaningful way” to the processes used by the AEC product to secure data being transmitted over a public network. *See* Ex. 1052 (Keromytis Decl.) at ¶¶30-31. That is incorrect.

71. There were two IP address-hopping schemes used by the AEC v3.0 product; namely, “Proxy Chaining” and the “MultiProxy” scheme.

72. The Aventail Proxy Chaining feature would forward connections for certain destinations through a succession of proxy servers. Ex. 1007 (Aventail) at



67-68. Each step or “hop” is an authenticated and encrypted link between the originating destination and the next destination. The overall path would ultimately connect the client computer to the final destination (e.g., the Extranet Server). Ex. 1007 (Aventail) at 63, 67.

73. The Aventail MultiProxy feature functioned in an analogous manner, allowing Aventail Connect to make connections or “hops” through successive proxy servers, where each formed an authenticated and encrypted link that connected the client computer to the final destination. Ex. 1007 (Aventail) at 63. In the MultiProxy scheme, the Aventail Connect client made connections to each proxy server in the chain individually, any or all of which could apply separate authentication, access control rules, and encryption, providing an additional level of security. Ex. 1007 (Aventail) at 63, 67.

74. Both the Proxy Chaining and MultiProxy features meaningfully contributed to the security of the VPNs established between client computers and the private network. In fact, the Proxy Chaining and Multi-Proxy features were included in the AEC v3.0 product to give administrators more control over the routing of traffic between a client and a private network to improve the overall security of their systems. Ex. 1007 (Aventail) at 67. For example, the Proxy Chaining capability would maintain an authenticated and encrypted tunnel between each proxy server and the Aventail Extranet Server. Ex. 1007 (Aventail) at 67-68.

The MultiProxy feature similarly would maintain authenticated and encrypted tunnels between a client computer and a secure target destination. Ex. 1007 (Aventail) at 67.

75. I also understand that VirnetX and its expert have claimed that the AEC v3.0 product does not “avoid[] sending a true IP address of the secure server to the client.” *See* Ex. 1052 (Keromytis Decl) at ¶ 32.

76. The Aventail Connect and AEC products worked by redirecting network traffic through one or more intermediary secure proxy servers. Because of this, the “true IP address” (e.g., the IP address of the host on the corporate network) is never sent back to a client – the communications are between the Extranet Server and the client running Aventail Connect.

77. In addition, Aventail Connect uses a locally stored hosts file to enable use of the Secure Extranet Explorer capability. This was a configuration file stored on the client that mapped a computer’s host names to its Windows machine name. Because the client already has the host names on the private network, there is no need to send the client computer the “true IP address” of the destination host machine. Ex. 1007 (Aventail) at 34.

78. I therefore disagree with the suggestion by VirnetX and its expert that the AEC v3.0 product sent the true IP address of a secure server back to the client.

## APPENDIX A

### MATERIALS CONSIDERED BY CHRIS HOPEN

Exhibit #	Reference Name
1007	Aventail Connect v3.01/2.51 Administrator's Guide and Aventail ExtraNet Server v3.0 Administrator's Guide (UNIX and Windows NT) (1996-1999)
1021	Aventail AutoSOCKS v2.1 Administration and User's Guide, 1996-1997
1022	Aventail Connect v3.1/v2.6 Administrator's Guide, 1996-1999
1051	Declaration of Jason Nieh, Ph.D., <i>Inter Partes</i> Reexamination Proceeding, Control No. 95/001,269 (April 15, 2010) (USPTO)
1052	Declaration of Angelos D. Keromytis, Ph.D., <i>Inter Partes</i> Reexamination Proceeding, Control No. 95/001,679 (July 20, 2012) (USPTO)
1058	"Aventail Ships the First Standards-Based Virtual Private Network Software Solution," PR Newswire, PR Newswire Association LLC, May 2, 1997
1059	Szeto, L., "Aventail delivers highly secure, flexible VPN solution," InfoWorld Media Group, June 23, 1997
1060	"Aventail Introduces the First Extranet-Ready Platform; Aventail Previews its Latest Solution, Aventail ExtraNet Center, at Networld+Interop in Atlanta," PR Newswire, PR Newswire Association LLC, October 12, 1998
1061	"Intranet Applications: Briefs," Network World, October 19, 1998

# EX. 1021

For Declaration of Chris Hopen

2.1

# Aventail AutoSOCKS

▲ ▲ ▲ **ADMINISTRATION  
& USER'S GUIDE**

**AVENTAIL**

Next Generation Security Systems



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

Copyright © 1996-1997 Aventail Corporation. All rights reserved.

117 South Main Street  
4<sup>th</sup> Floor  
Seattle, WA 98104-2540  
USA

Printed in the United States of America.

## **Trademarks and Copyrights**

Aventail, AutoSOCKS, Internet Policy Manager, Aventail VPN, Mobile VPN, and Partner VPN are trademarks of Aventail Corporation.

Socks5Toolkit is a trademark of NEC Corporation. MD4 Message-Digest Algorithm and MD5 Message-Digest Algorithm are trademarks of RSA Data Security, Inc. Microsoft, MS, Windows, Windows 95, and Windows NT are either registered trademarks or trademarks of Microsoft Corporation. RealAudio is a trademark of Progressive Networks.

Other product names mentioned in this manual may be trademarks or registered trademarks of their respective companies and are the sole property of their respective manufacturers.

Copyright © 1995-1996 NEC Corporation. All rights reserved.

Copyright © 1990-1992, RSA Data Security, Inc. All rights reserved.

Copyright © 1991-1992, RSA Data Security, Inc. All rights reserved.

# Table of Contents

<b>Introduction.....</b>	<b>1</b>
About This Document .....	1
Document Organization.....	2
Document Conventions .....	2
Technical Support.....	3
About Aventail Corporation .....	4
<b>AutoSOCKS v2.1 Administration and User's Guide.....</b>	<b>5</b>
Getting Started.....	5
Network Security in a Nutshell.....	5
What is AutoSOCKS?.....	6
TCP/IP Communications .....	6
WinSock Connection to A Remote Host .....	6
What Does AutoSOCKS Do?.....	7
AutoSOCKS Platform Requirements.....	9
Windows 95 and Windows NT 4.0 .....	9
System Requirements .....	9
Interface Features .....	9
Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51 .....	10
System Requirements .....	10
Interface Features .....	10
Installation Source Media .....	10
Installing AutoSOCKS .....	11
Configuration Files.....	11
Individual Installation .....	11
Network Installation.....	13
Networked Configuration File Setup.....	14
Administrator-Maintained Shared Configuration Files .....	14
Shared Configuration File Distribution .....	14
Setup Command Line Options .....	15
Configuring AutoSOCKS .....	16
Define a SOCKS Server .....	18
Define a Destination.....	20

Enter Redirection Rules.....	23
Define Local Name Resolution.....	26
Managing Authentication Modules.....	27
Example Network Configurations .....	35
Configuration Using Aventail Internet Policy Manager .....	36
Configuration Using Aventail VPN Server .....	37
<b>AutoSOCKS Utilities Reference Guide .....</b>	<b>42</b>
System Menu Commands .....	42
Close.....	43
Hide Icon .....	43
Help.....	43
About.....	43
Credentials .....	43
Configuration File.....	44
Config Tool.....	45
Logging Tool .....	46
S5 Ping.....	51
<b>AutoSOCKS User Supplement.....</b>	<b>55</b>
How to Start and Close AutoSOCKS.....	55
How to Enter Authentication Credentials .....	56
Username/Password and CHAP Authentication .....	57
SSL Authentication.....	58
Appendix I: Troubleshooting.....	61
AutoSOCKS Installation Problems .....	61
Network Connectivity Problems .....	62
AutoSOCKS Configuration Problems .....	62
Application and TCP/IP Stack Interoperability Problems .....	64
AutoSOCKS Trace Logging.....	64
Reporting AutoSOCKS Problems.....	68
Glossary.....	70
Index.....	72



# 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.

<b>Convention</b>	<b>Usage</b>
ALL CAPITALS	Filenames and extensions, directory names, keynames, and pathnames.
<b>Bold</b>	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 ( <b>Destination</b> field), e-mail addresses ( <a href="mailto:support@aventail.com">support@aventail.com</a> ), URLs ( <a href="http://www.aventail.com/">http://www.aventail.com/</a> ), and IP addresses ( <b>165.121.6.26</b> ) are also bold.
<i>Italic</i>	Placeholders that represent information the user must insert.
“To Do” Procedures	Underlined <i>To Do</i> 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](mailto: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

4<sup>th</sup> Floor

Seattle, WA 98104-2540

Phone: 206.777.5600

Fax: 206.777.5656

<http://www.aventail.com/>

[info@aventail.com](mailto: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](mailto: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.)

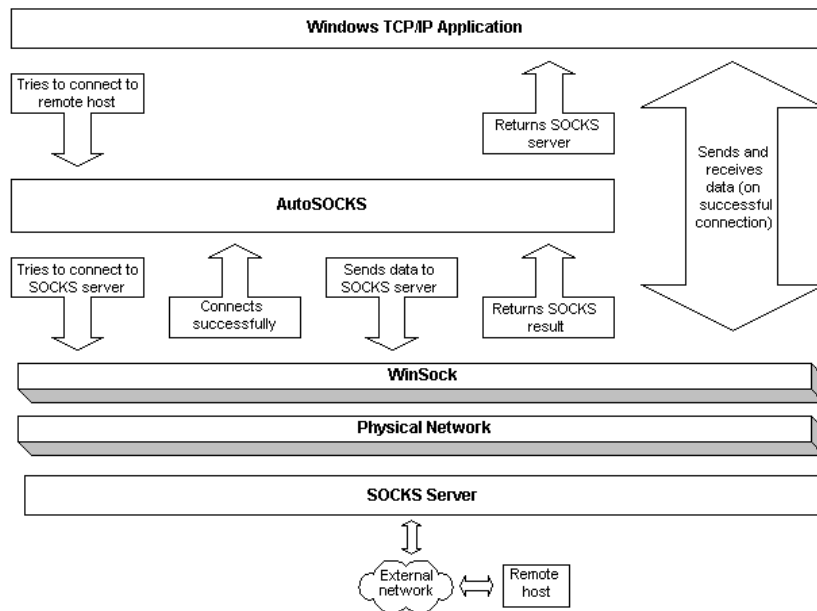


Figure 1. Network application calls intercepted by AutoSOCKS

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.

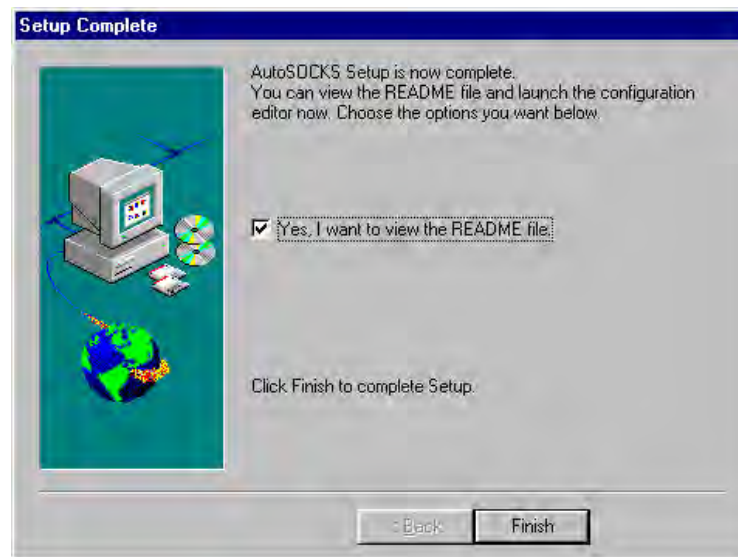
- If you are installing from a network-delivered self-extracting archive, simply run the archive file. This will extract the installation files and automatically launch the setup program.

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

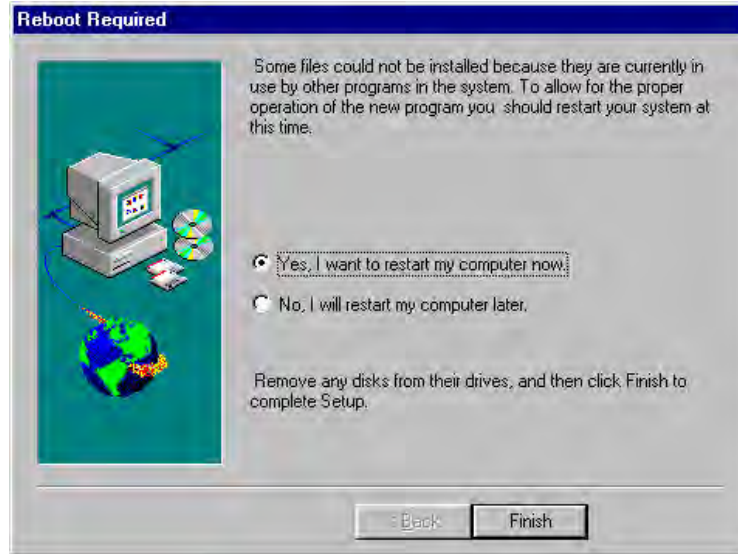
2. At the end of the Setup Program you can click the **Yes, I want to view the README file** box in the Setup Complete dialog box. This opens the README file for the latest information on AutoSOCKS.

-OR-

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



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



- After restarting your PC, start AutoSOCKS for the first time.
- 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.
- 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 UNC.)

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= <i>path</i>	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= <i>path</i>	Specifies the directory containing AutoSOCKS installation files.	<b>C:\Program Files\Aventail\AutoSOCKS</b>	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-

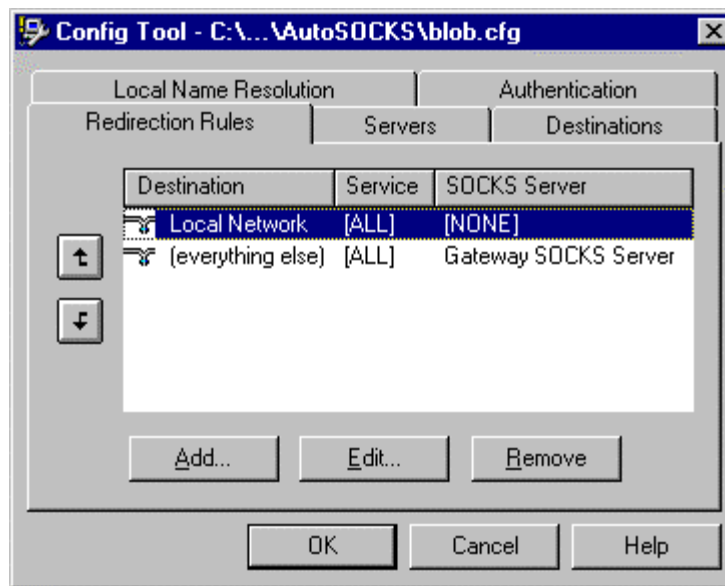
Select Config Tool from the Aventail AutoSOCKS program group (Windows 3.1, Windows for Workgroups 3.11, or Windows NT 3.51) or the Aventail AutoSOCKS menu (Windows 95 or Windows NT 4.0 Programs menu option).

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.



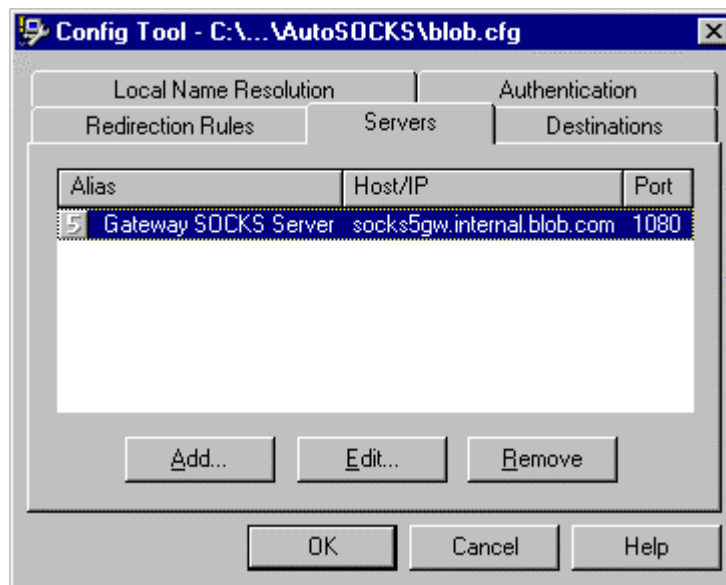
The Config Tool window contains five tabs. The properties defined on each tab can be edited at any time.

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.

**Define SOCKS Server**

Alias Name: Gateway SOCKS Server

Hostname or IP: socks5gw.internal.blob.com

Port Number: 1080

**SOCKS Version**

SOCKS v4

SOCKS v5

Detect Version

**Fallback**

Fallback to Server: Gateway SOCKS Server

Fallback to Host Alias

OK Cancel Help

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

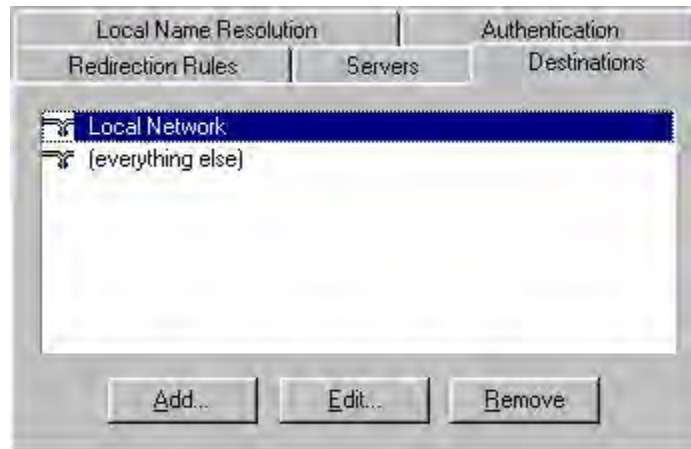
---

- 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.

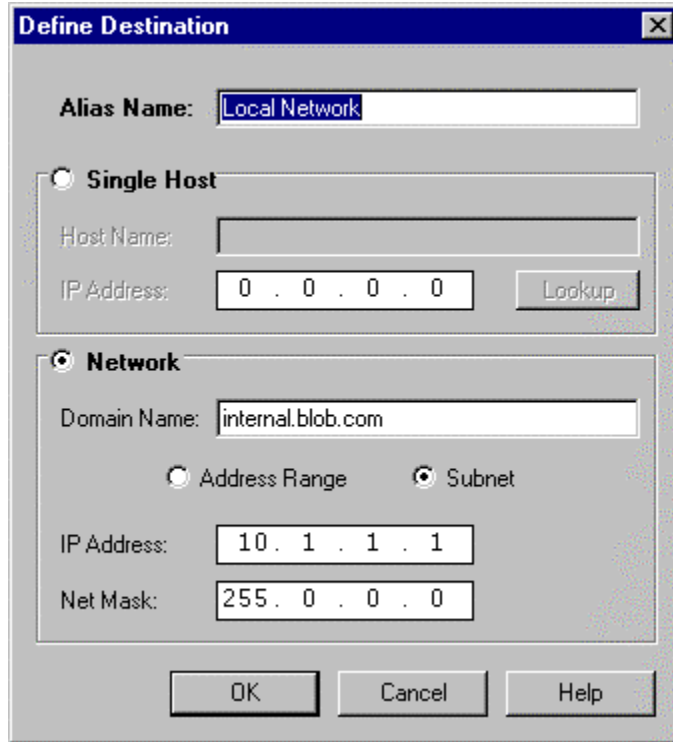
Note: The **(everything else)** destination refers to all network and host addresses not otherwise defined.

#### 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

---

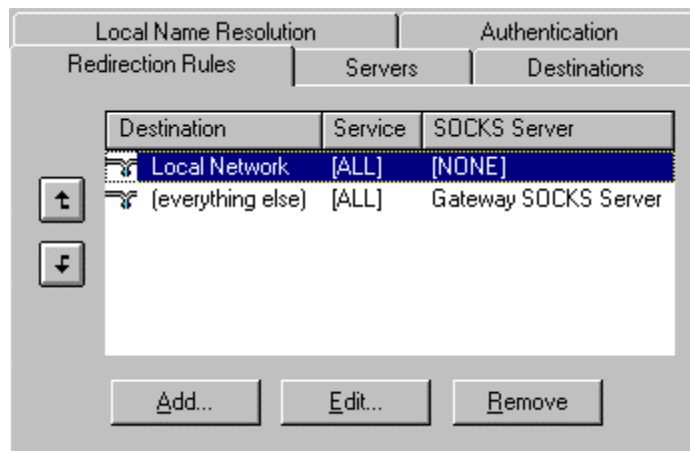
- Select the destination you want to remove and click the **Remove** button.

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

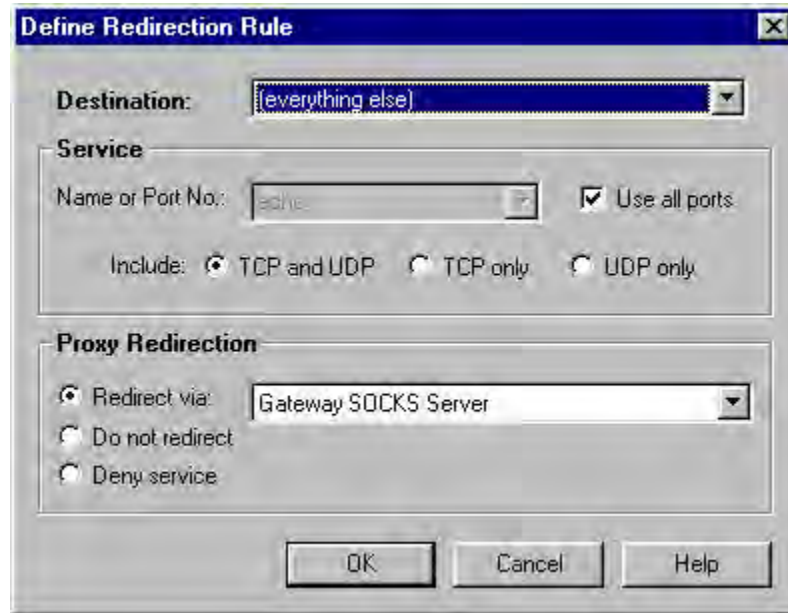
As you add destinations, use the arrow buttons to prioritize them. List the most specific rules first and the general rules last.

*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.

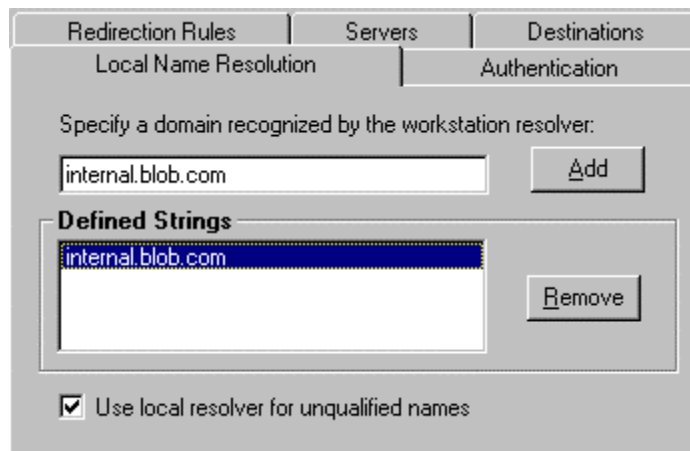
### Define Local Name Resolution

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 reenter 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.

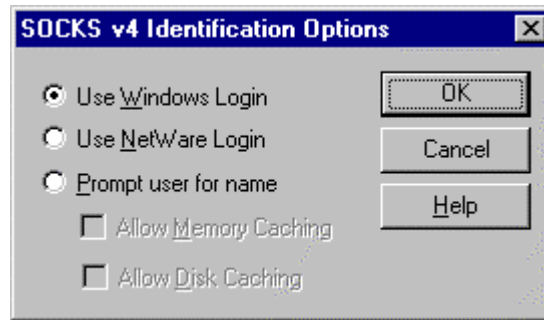
To configure the SOCKS v4 Identification module

---

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**.

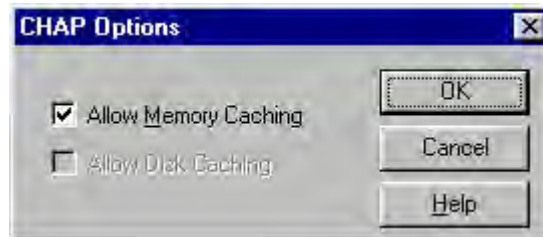
The dialog box closes and the Config Tool is displayed.

#### 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.

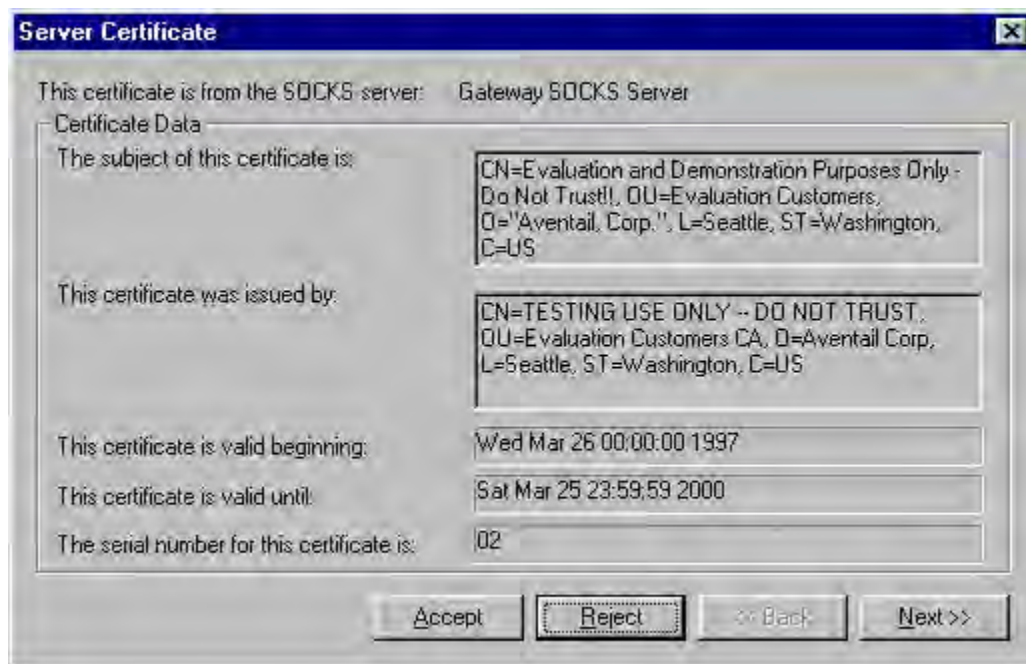
#### To configure the SSL security module

---

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.

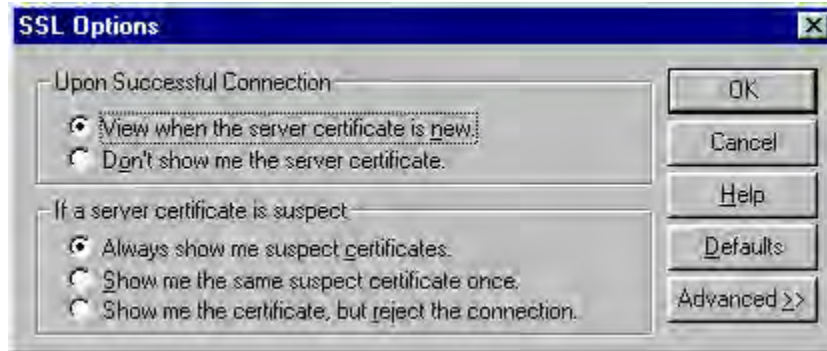
Certificates contain special integrity checks and electronic signatures which verify that the certificate is genuine, was issued by some certification authority, and was not tampered with. Anybody can issue a certificate that says anything; the client must know who issued the certificate, and have some 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 **sslcnt** (SSL Security) and click the **Setup** button.

The SSL Options dialog box appears.





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

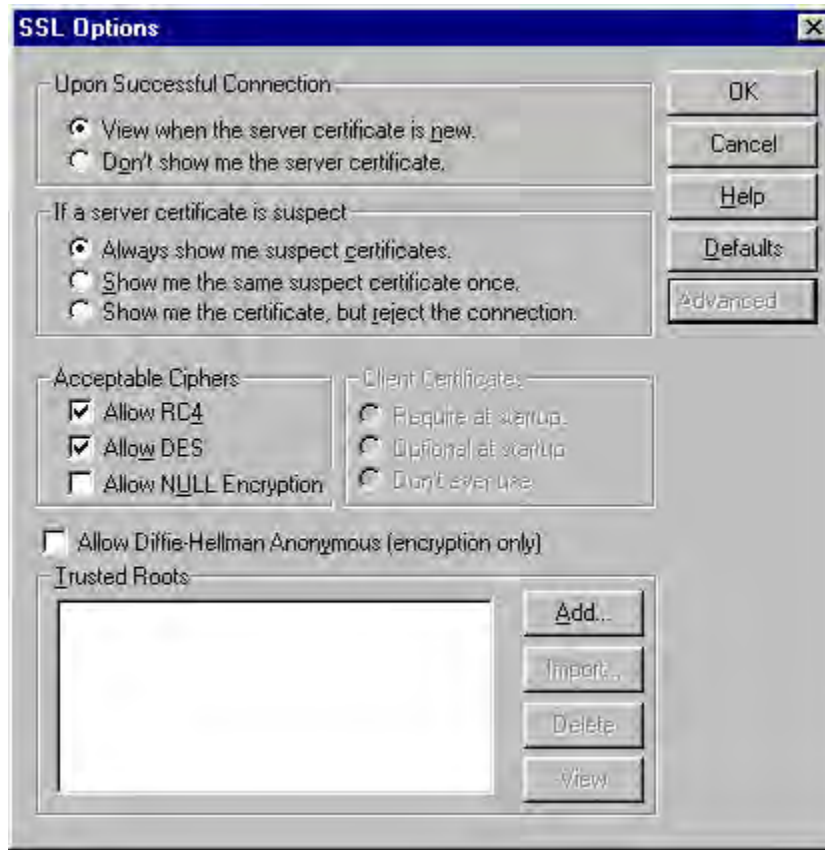
Is the certificate valid?

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.

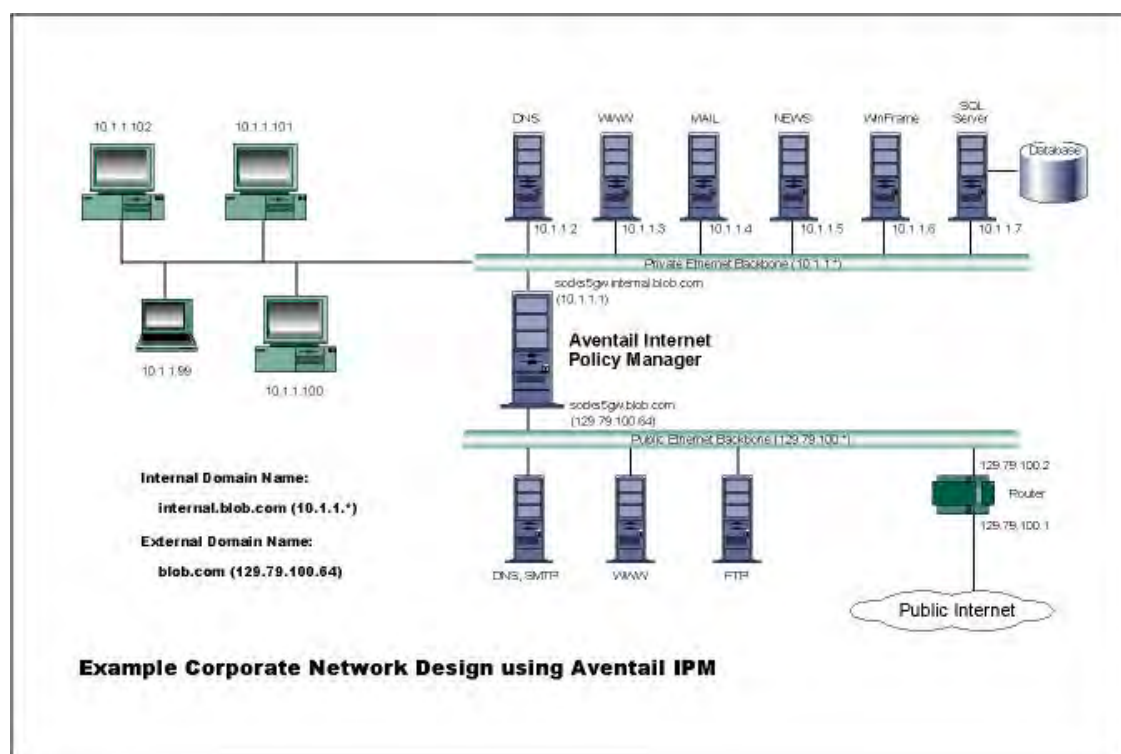
The dialog box closes and the Config Tool is displayed.

## 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.

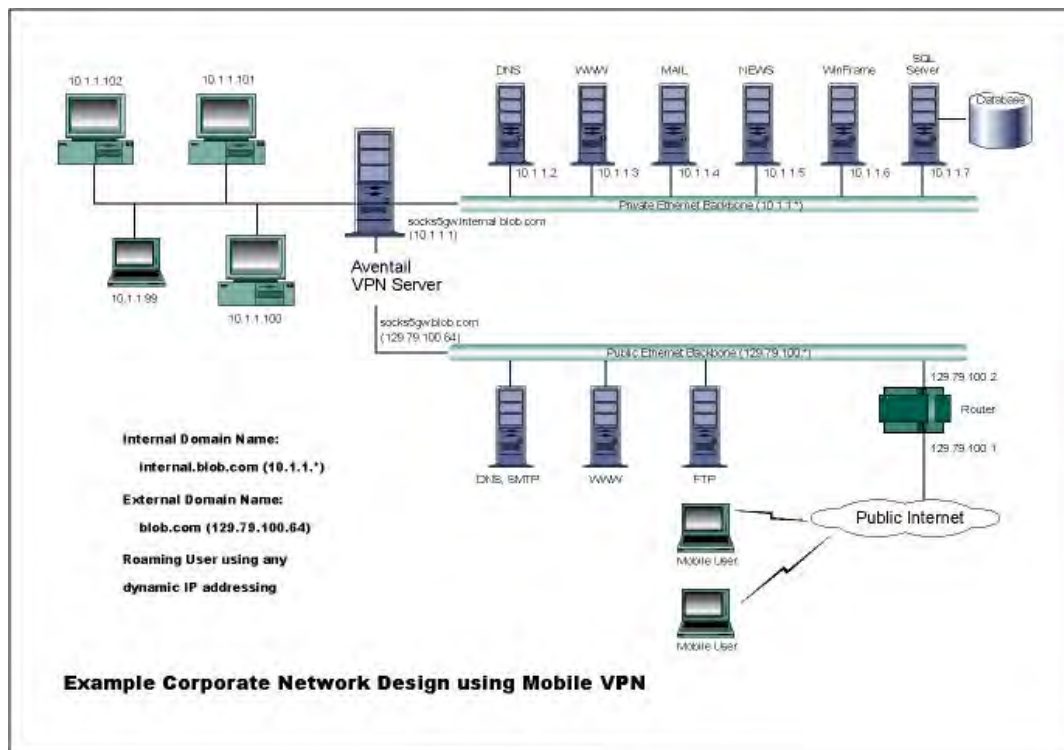
The other servers on the private segment are "internal" or private servers that contain information and tools that are not intended for public use or consumption. If these individual hosts require access beyond the Aventail IPM they can also be configured to use AutoSOCKS. As in the client workstation 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.



### 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 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."

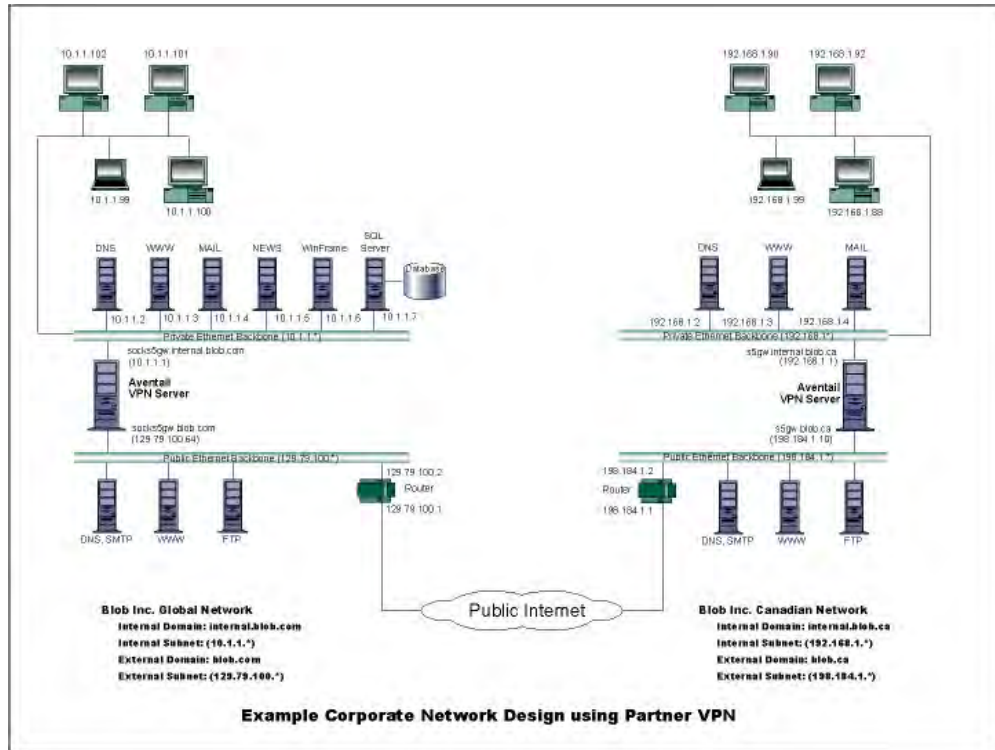


Figure 4. AutoSOCKS in a Partner VPN Environment





# 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.

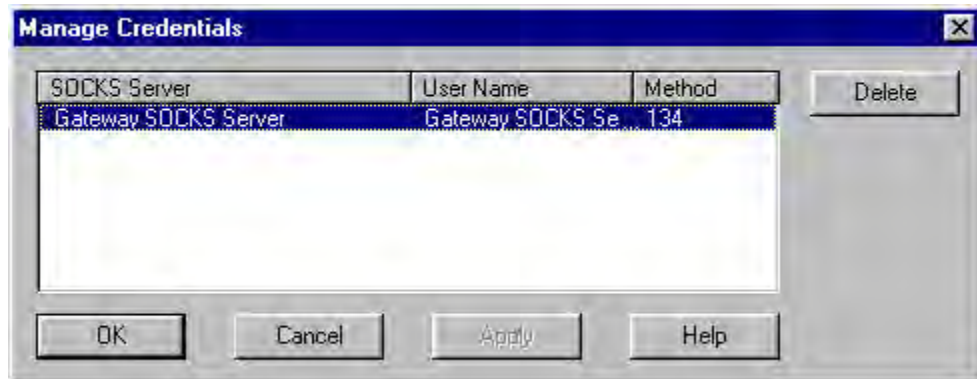
## About

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.

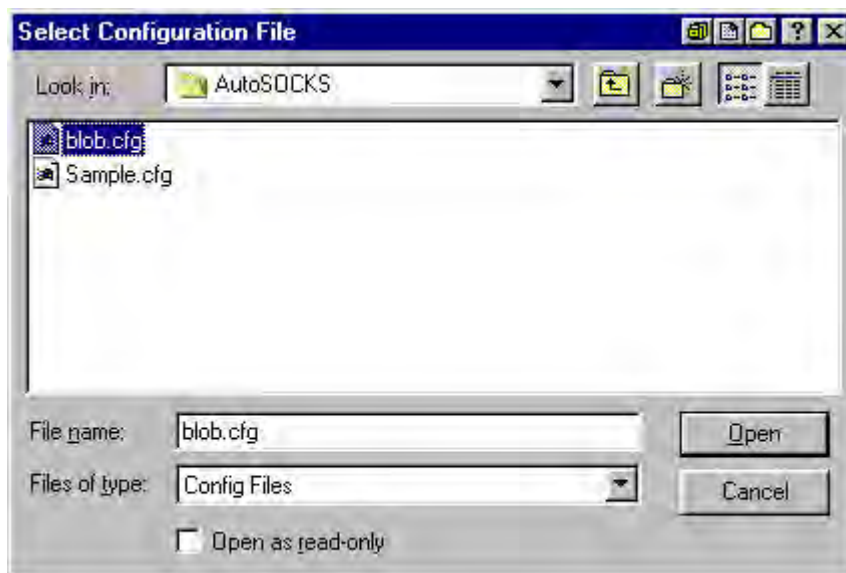
-OR-

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

---

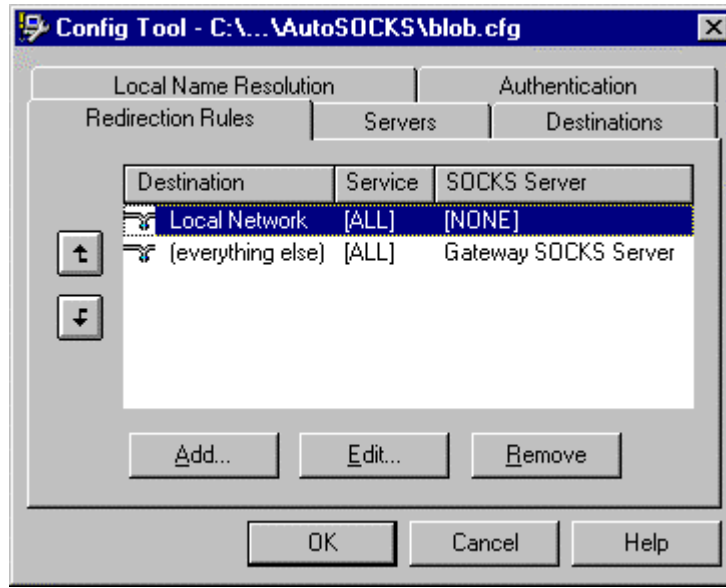
Check with the network administrator before making any changes to the configuration.

- 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.)



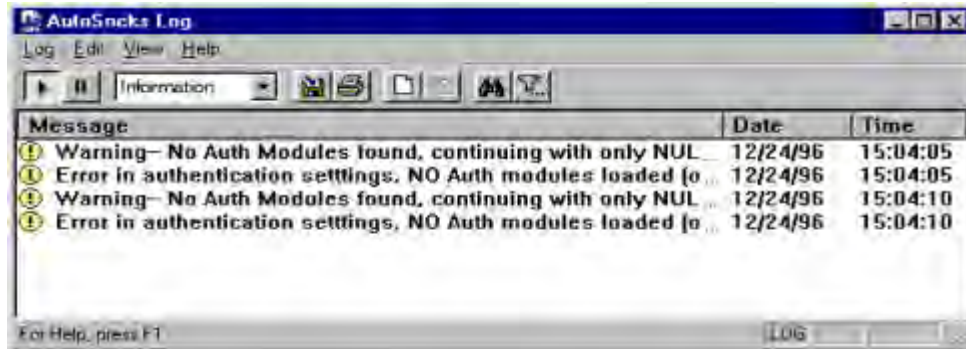
Configuration files should be set up by a network administrator. They are usually created during AutoSOCKS installation but they can also be added, removed, or modified at any time. If necessary, several configuration files can be created for different users or user groups. Some configuration files may 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.



- 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

- 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.

- 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

---

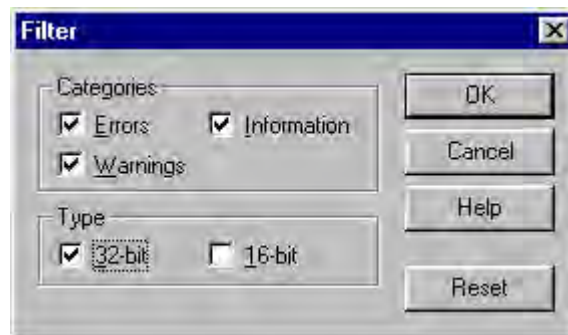
The contents of a log window can be filtered by selecting the types of messages you wish to view. Selecting a specific type of message can make it easier to scan the information onscreen. If the data has been saved to a log file, a view filter will not affect the file contents; it merely adjusts the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





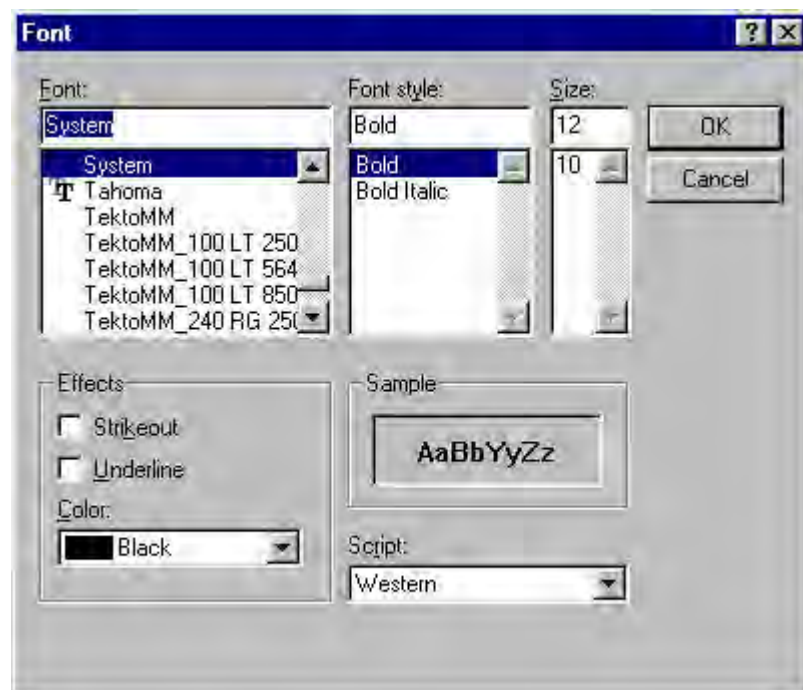
Field	Definition	
Categories	Select any of the three filters to display errors, warnings, and/or information in the log window.	
Type*	32-bit:	Show messages from 32-bit applications.
	16-bit:	Show messages from 16-bit applications.
	*These options are disabled if you're running 16-bit Windows.	

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**.

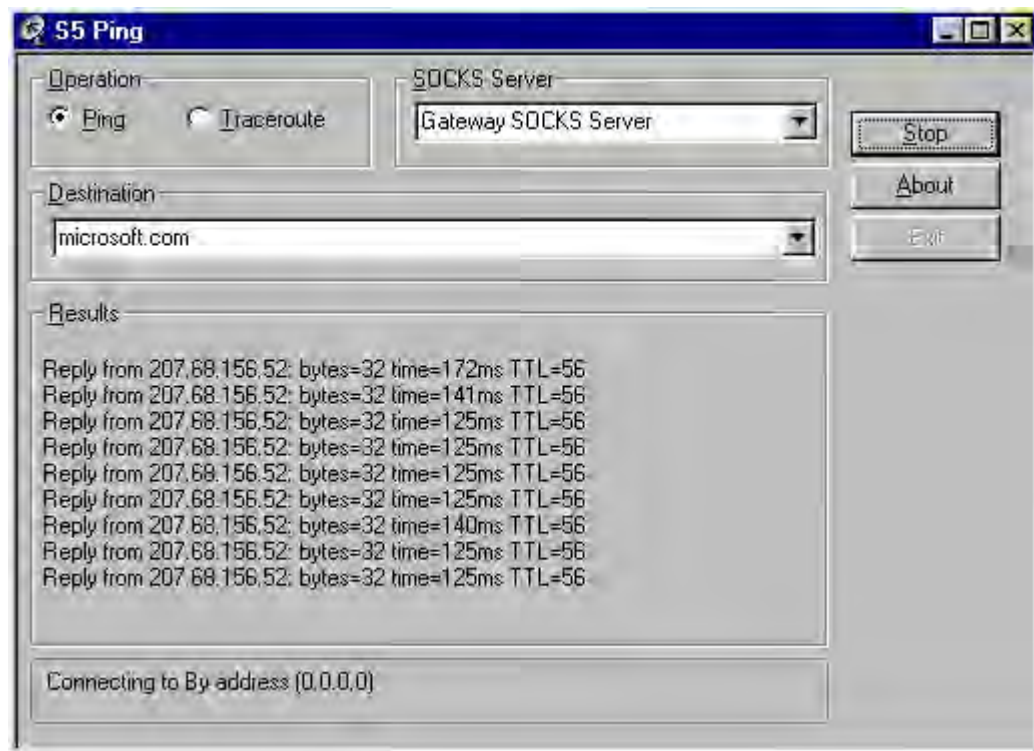
## S5 Ping

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.

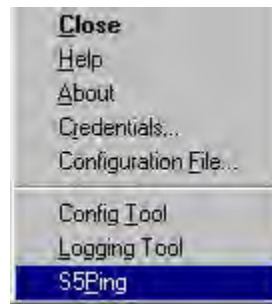
You'll see a minimized AutoSOCKS icon indicating that AutoSOCKS is running in the background. In Windows 95 and Windows NT 4.0, this icon is located in the system tray on the Task bar.



### 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**.

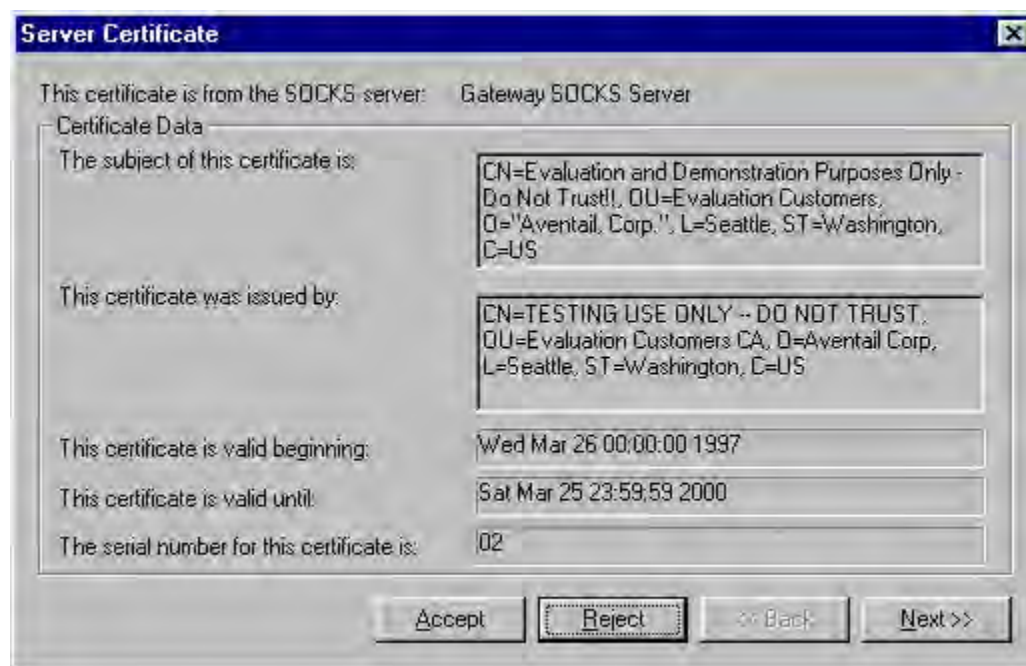
When you click OK, your credentials are sent to the secure SOCKS server and if they are accepted, you'll continue your processing without hindrance.

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.

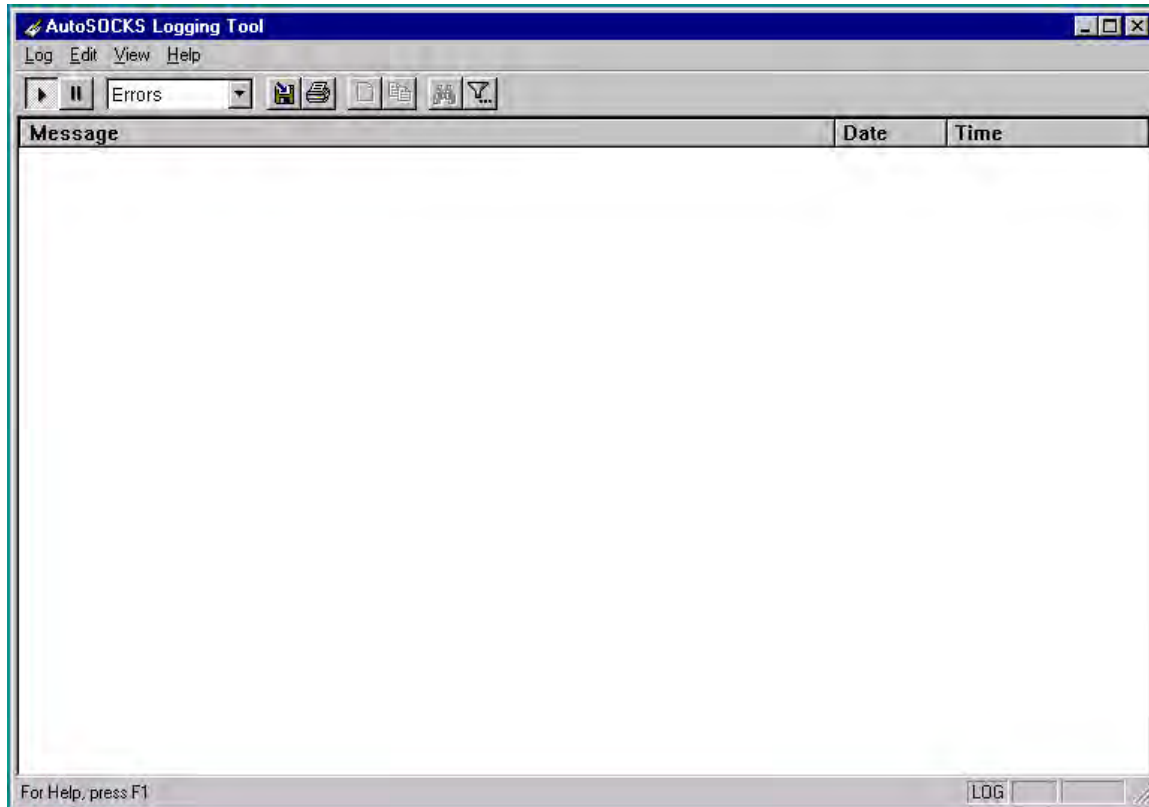


Figure 1

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.

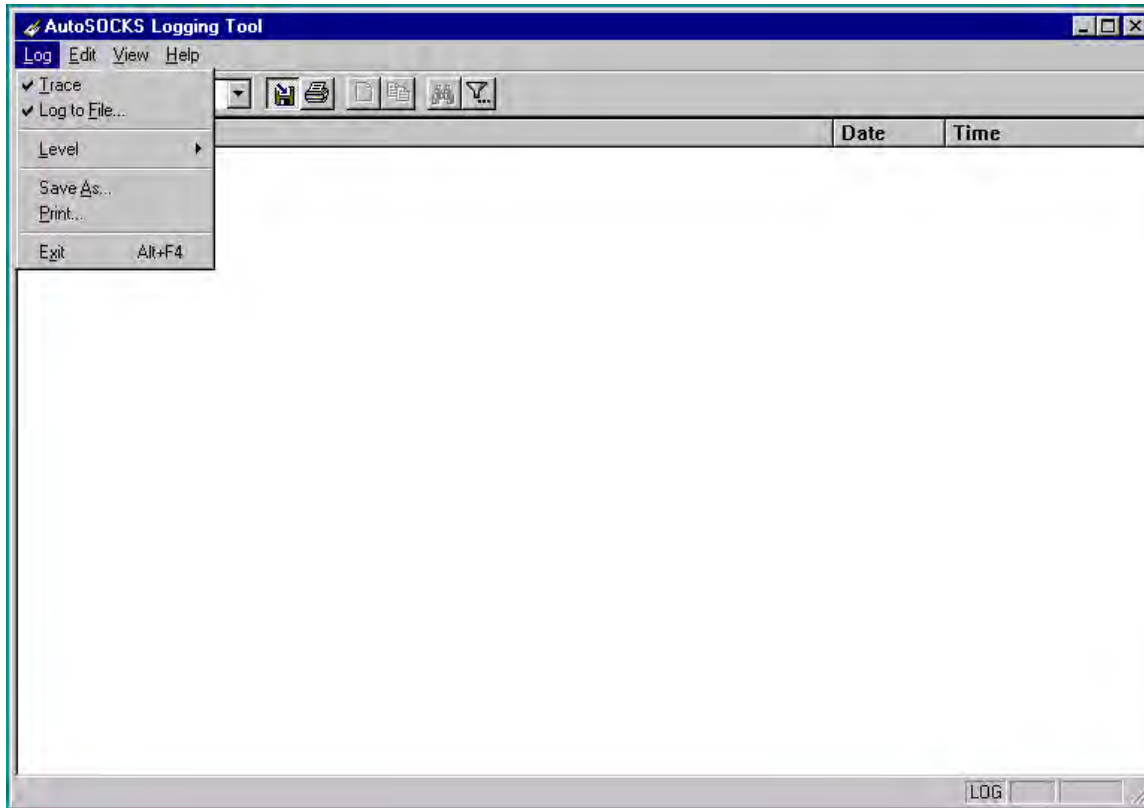


Figure 2

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.

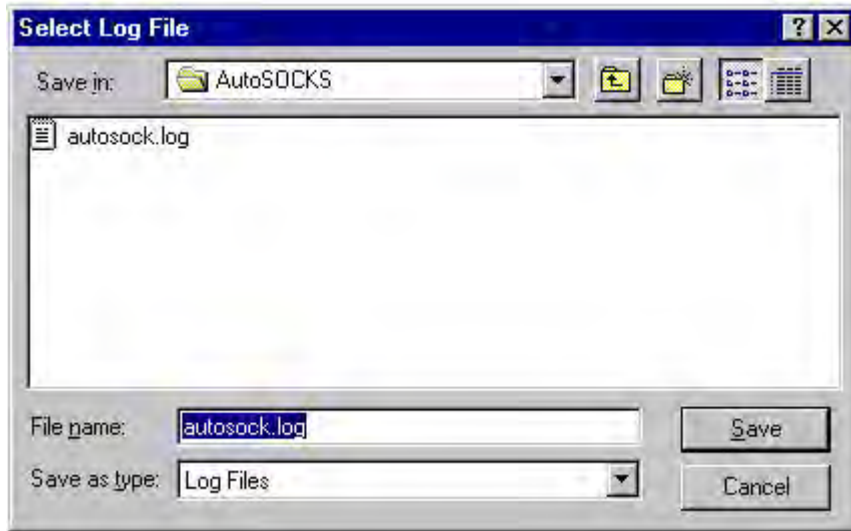


Figure 3

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.

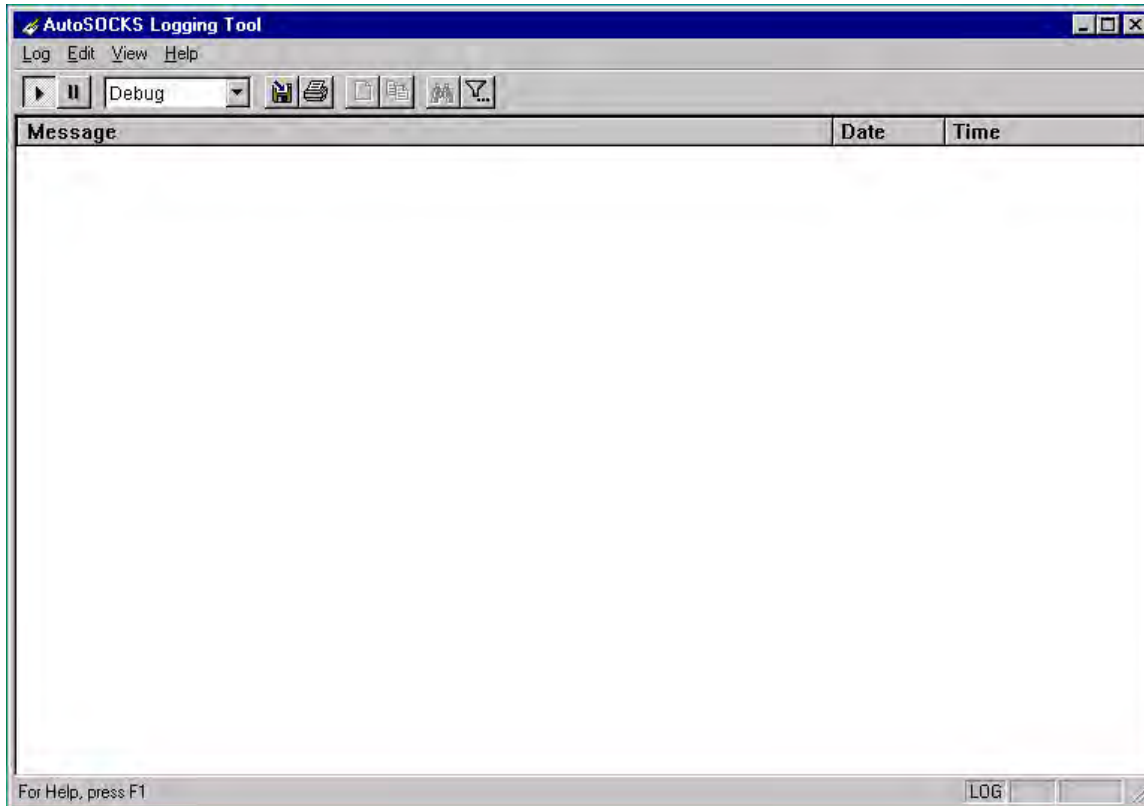


Figure 4

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 its 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

About .....	42	install .....	11
About command.....	43	menu commands .....	42
About This Document		platforms .....	9
conventions .....	2	requirements.....	9, 10
organization .....	2	setup .....	11
Address Range.....	23	source media .....	10, 11
Administrator's Guide .....	5	starting and closing .....	55
Administrator-Maintained		system requirements.....	9, 10
Shared Configuration		User Supplement.....	55
Files .....	14	what does it do .....	7
Alias.....	19, 20, 23	what is it .....	6
authentication		AutoSOCKS in a Partner	
CHAP.....	30	VPN Network .....	40
managing modules .....	27	AutoSOCKS in an Aventail	
SOCKS V4 .....	29	IPM Environment.....	36
SSL .....	31	AutoSOCKS in an Aventail	
Username/Password.....	29	Mobile VPN	
Authentication.....	6	Environment.....	38
credentials .....	43	Aventail Corporation .....	4
AutoSOCKS		CHAP .....	44
network installation.....	13	CHAP authentication .....	30
uninstall .....	13	Close .....	42
AutoSOCKS		Close command .....	43
About command.....	43	closing AutoSOCKS .....	56
Close command.....	43	Config Tool .....	42
Configuration File		Configuration file	
command.....	44	distribution .....	14
Credentials command .....	43	network .....	14
getting started.....	5	shared.....	14
Help command .....	43	Configuration File .....	42
Hide Icon command .....	43		



Configuration File	command.....44
Configuration Files .....	11
Configuring AutoSOCKS .....	45
Credentials .....	42, 43
delete .....	44
exit dialog box.....	44
Define a Destination .....	20
Define a SOCKS Server.....	18
Destination	
add .....	21
define .....	20
remove.....	23
Encryption.....	6
Enter Redirection Rules .....	23
Features of AutoSOCKS .....	1
filter messages .....	48
Getting Started .....	5
Glossary .....	70
Hardware Requirements .....	9, 10
Help .....	42
Help command.....	43
Hide Icon.....	42
Hide Icon command.....	43
How to Enter	
Authentication	
Credentials .....	56
Installation Source Media .....	10, 11
Installing AutoSOCKS .....	11
Interface Features .....	9, 10
Introduction.....	1
IPM Environment .....	36
Local Name Resolution.....	18, 26
Log File	
clear.....	50
close .....	50
copy.....	49
filter.....	48
find.....	50
print.....	50
save .....	47
view parameters .....	49
Logging Tool.....	42, 46
Managing Authentication	
Modules .....	27
Network Installation .....	13
Network Security in a	
Ntshell.....	5
Networked Configuration	
File Setup .....	14
Ping42, 52	
Platform Requirements .....	9
procedures	
To accept an SSL	
certificate.....	58
To add a destination .....	21
To add a local domain	
name.....	27
To add a redirection rule .....	24
To add a SOCKS server .....	19
To change the view	
parameters.....	49
To clear the log window .....	50
To close AutoSOCKS .....	56
To close the log window .....	50

To configure the CHAP Authentication module .....	30	To stop Ping or Traceroute and close S5 Ping .....	53
To configure the SOCKS v4 authentication module .....	29	To trace AutoSOCKS activity .....	46
To configure the SSL security model .....	31	To uninstall AutoSOCKS .....	13
To configure the Username/Password authentication module .....	29	redirection rules	
To copy the log window .....	49	add .....	24
To delete a credential entry .....	44	enter .....	23
To distribute a shared configuration file .....	14	remove .....	26
To edit a destination .....	23	S5 Ping	
To edit a redirection rule .....	26	Ping .....	42
To edit SOCKS server properties .....	20	Traceroute .....	42
To enter authentication credentials .....	57	S5 Ping .....	51
To exit the Manage Credentials dialog box .....	44	Setup Command Line Options .....	15
To filter messages in the log window .....	48	Shared Configuration File Distribution .....	14
To find a specific message .....	50	SOCKS Server	
To install AutoSOCKS .....	11	add .....	19
To launch S5 Ping .....	52	define .....	18
To launch the Config tool .....	17	remove .....	20
To load a configuration file .....	45	SOCKS V4 authentication .....	29
To print the log window .....	50	socksification .....	6
To remove a local name .....	27	SSL 58	
To remove a redirection rule .....	26	SSL authentication .....	31, 58
To remove a SOCKS server definition .....	20	Stardust WinSock Labs .....	1
To run Ping or Traceroute using S5 Ping .....	53	Starting and Closing AutoSOCKS .....	55
To save a log file .....	47	starting AutoSOCKS .....	55
To start AutoSOCKS .....	55	Subnet .....	23
		System menu	
		About command .....	43
		Close command .....	43
		commands .....	42
		Credentials command .....	43

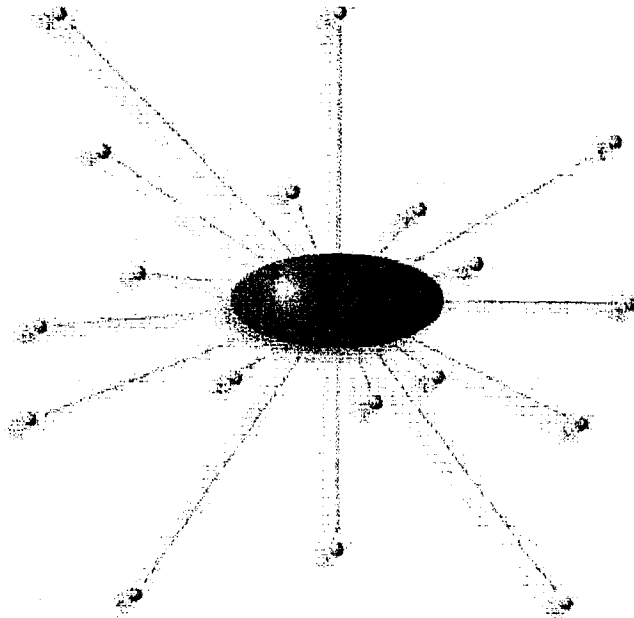
Help command .....	43	User Supplement.....	55
Hide Icon command .....	43	Username/Password and	
TCP/IP Communications .....	6	CHAP Authentication .....	57
Technical Support .....	3	Username/Password	
trace		authentication.....	29
Logging tool.....	46	VPN Environment.....	38
Traceroute .....	42, 52	VPN Partner Network.....	40
Troubleshooting .....	61	What is AutoSOCKS? .....	6
UDP .....	6, 25, 71		

# EX. 1022

For Declaration of Chris Hopen

# Aventail CONNECT

v3.1/v2.6



Administrator's Guide

Windows



# AVENTAIL CONNECT 3.1/2.6 ADMINISTRATOR'S GUIDE

© 1996-1999 Aventail Corporation. All rights reserved.

808 Howell Street, Second Floor  
Seattle, WA 98101  
USA

<http://www.aventail.com/>

Printed in the United States of America.

## TRADEMARKS AND COPYRIGHTS

Aventail is a registered trademark of Aventail Corporation. AutoSOCKS, Internet Policy Manager, Aventail VPN, Aventail VPN Client, Aventail ExtraNet Center, and Aventail ExtraNet Server are trademarks of Aventail Corporation.

Socks5Toolkit is a trademark of NEC Corporation. MD4 Message-Digest Algorithm and MD5 Message-Digest Algorithm are trademarks of RSA Data Security, Inc. Microsoft, MS, Windows, Windows 95, Windows 98, and Windows NT are either registered trademarks or trademarks of Microsoft Corporation. RealAudio is a trademark of RealNetworks. SecurID, SoftID, ACE/Server, and SDTI are either registered trademarks or trademarks of Security Dynamics Technologies, Inc.

This product includes software written by Dr. Stephen Henson.

Other product names mentioned in this manual may be trademarks or registered trademarks of their respective companies and are the sole property of their respective manufacturers.

© 1995-1996 NEC Corporation. All rights reserved.

© 1990-1992 RSA Data Security, Inc. All rights reserved.

© 1996 Hi/fn Inc., including one or more U.S. patents: 4701745, 5016009, 5126739, and 5146221, and other patents pending.

© 1996-1997 Consensus Development Corporation. All rights reserved.

**Table of Contents**

**TROUBLESHOOTING**

    Trademarks and Copyrights . . . . . i

**INTRODUCTION** . . . . . 1

    About This Document . . . . . 3

    Document Organization . . . . . 3

    Document Conventions . . . . . 4

    Aventail Technical Support . . . . . 5

    About Aventail Corporation . . . . . 5

**ADMINISTRATOR'S GUIDE**

    Getting Started . . . . . 6

        Network Security in a Nutshell . . . . . 6

        What is Aventail Connect? . . . . . 7

        What Does Aventail Connect Do? . . . . . 9

        How Does Aventail Connect Work? . . . . . 11

        Aventail Connect Platform Requirements . . . . . 13

        Interface Features . . . . . 14

        Installation Source Media . . . . . 14

    Installing Aventail Connect . . . . . 15

        Configuration Files . . . . . 15

        Customized Configuration and Distribution . . . . . 16

        Individual Installation . . . . . 16

        Network Installation . . . . . 18

        Administrative Setup . . . . . 21

        Customizer . . . . . 22

    Configuring Aventail Connect . . . . . 33

        Define an Extranet (SOCKS) Server . . . . . 35

        Define a Destination . . . . . 39

        Enter Redirection Rules . . . . . 42

        Define Name Resolution . . . . . 45

        Manage Authentication Modules . . . . . 46

        Advanced Tab Options . . . . . 62

        Enable Password Protection . . . . . 67

        Multiple Firewall Traversal . . . . . 68

    Example Network Configuration . . . . . 76

        Configuration Using Aventail ExtraNet Server . . . . . 76

**UTILITIES REFERENCE GUIDE**

System Menu Commands . . . . . 80  
     Close . . . . . 80  
     Hide Icon . . . . . 81  
     Help . . . . . 81  
     About . . . . . 81  
     Credentials . . . . . 81  
     Configuration File . . . . . 82  
 Utilities . . . . . 83  
     Config Tool . . . . . 84  
     Logging Tool . . . . . 84  
     S5 Ping . . . . . 92  
 Secure Extranet Explorer . . . . . 95  
     How Extranet Neighborhood Works . . . . . 96  
     Installing Extranet Neighborhood . . . . . 97  
     Configuring Extranet Neighborhood . . . . . 97  
     SEE Properties . . . . . 101

**TROUBLESHOOTING**

Aventail Connect Installation Problems . . . . . 107  
 Network Connectivity Problems . . . . . 108  
 Aventail Connect Configuration Problems . . . . . 108  
 Application and TCP/IP Stack Interoperability Problems . . . . . 110  
 Aventail Connect Trace Logging . . . . . 110  
 Error Messages . . . . . 111  
 Reporting Aventail Connect Problems . . . . . 112

**GLOSSARY . . . . . 113**

**INDEX . . . . . 117**



# Introduction

Welcome to the Aventail Connect 3.1/2.6 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 (LSP) applications in Windows 98, and Windows NT 4.0, and WinSock 1.1 and WinSock 2 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



**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.
<b>Bold</b>	Dialog box controls ( <b>Edit...</b> buttons), e-mail addresses ( <b>support@aventail.com</b> ), URLs, ( <b>www.aventail.com</b> ), and IP addresses ( <b>165.121.6.26</b> ).
<i>Italic</i>	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](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](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](mailto: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](mailto:info@aventail.com)



An aventail is a piece of chainmail armor worn around the neck area. In the 14<sup>th</sup> 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 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](mailto: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.
- 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 (Windows sockets) 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 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**

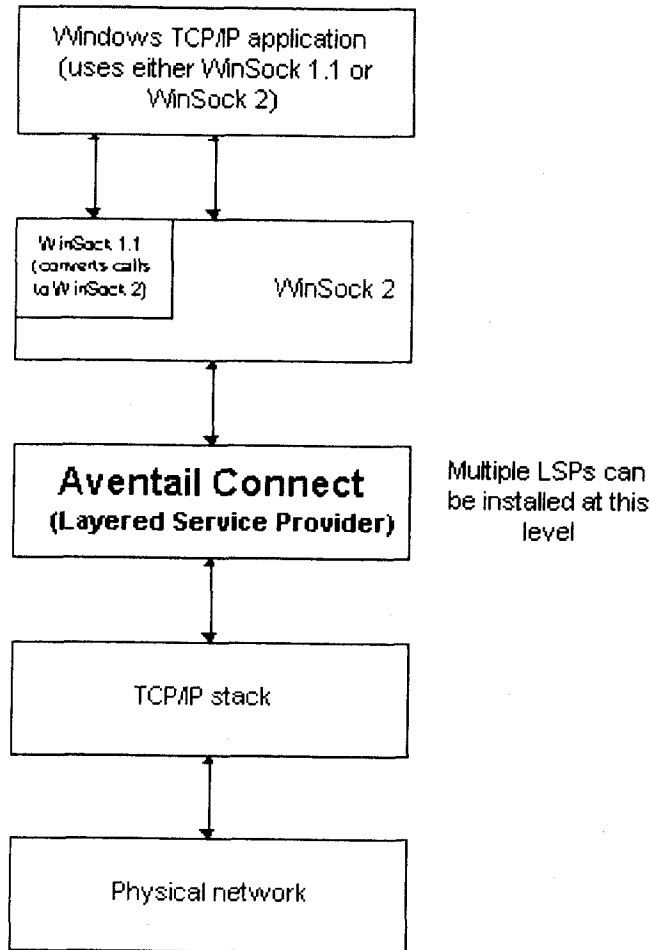
Via WinSock, an application goes through the following steps to connect to a remote host on the Internet or corporate extranet:

1. The application executes a Domain Name System (DNS) lookup to convert the hostname into an Internet Protocol (IP) address or, in rare cases, it will do a reverse DNS lookup to convert the IP address into a hostname. 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 3.1 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 versions 1.1 and 2. Aventail Connect 3.1, like all LSPs, requires WinSock 2; WinSock 1.1 does not support LSPs. WinSock 2 includes backward-compatibility with all WinSock 1.1 applications. Not every platform supports WinSock 2 and its LSP structure.

- Windows 98 and Windows NT 4.0 support WinSock 2 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, but you must install a patch (available from Microsoft) to add support for WinSock 2.
- Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51 do not support WinSock 2; they support only WinSock 1.1.

For those platforms that do not support WinSock 2 and LSP applications, Aventail includes Aventail Connect 2.6 on the Aventail Connect 3.1/2.6 CD. Aventail Connect 2.6 was designed for operating systems that support only WinSock 1.1. On Windows 3.1, Windows for Workgroups 3.11, or Windows NT 3.51 operating systems, setup will install Aventail Connect 2.6. If you are working on a Windows 95 operating system, setup will detect whether you have installed the Microsoft Windows 95 WinSock 2 Update. If setup detects the Microsoft update, which upgrades Windows 95 to support WinSock 2, setup will install Aventail Connect 3.1. If setup does not detect the Microsoft update, it will install Aventail Connect 2.6.

The Aventail Connect 2.6 user interface is identical to that of Aventail Connect 3.1; however, Aventail Connect 3.1 includes MultiProxy functionality (see "Multiple Firewall Traversal"). Aventail Connect 2.6 does not include MultiProxy.

In the future, more Windows applications may require WinSock 2.

During installation, setup determines which version of Aventail Connect to install. On WinSock 2 platforms, Aventail Connect 3.1 is installed. On WinSock 1.1 platforms, Aventail Connect 2.6 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	Aventail Connect 3.1
Windows 95	With Microsoft patch: WinSock 2	Aventail Connect 3.1
	Without Microsoft patch: WinSock 1.1	Aventail Connect 2.6
Windows 3.1, Windows for Workgroups 3.11, Windows NT 3.51	WinSock 1.1	Aventail Connect 2.6

You can create custom packages that include one or both versions of Aventail Connect (3.1 and 2.6). 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 Update upgrades WinSock 1.1 to WinSock 2 in Windows 95. This patch (filename `w95ws2setup.exe`) is available from the Microsoft Web site, at [http://www.microsoft.com/Windows95/downloads/contents/wuadmintools/s\\_wunetworkingtools/W95Sockets2/default.asp](http://www.microsoft.com/Windows95/downloads/contents/wuadmintools/s_wunetworkingtools/W95Sockets2/default.asp). Unless you need specific Aventail Connect 3.1 features, Aventail recommends that you do not upgrade from WinSock 1.1 to WinSock 2. If you do not upgrade to WinSock 2, Aventail Connect 2.6 will be installed on Windows 95 systems.

If you do need to install the Microsoft Windows 95 WinSock 2 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 or, in rare cases, it will do a reverse DNS lookup to convert the IP address to a hostname. 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 false 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.



**CAUTION:** *The reverse DNS process can create unexpected delays, causing Aventail Connect to behave unpredictably. Aventail recommends that you do not enable this option unless you specifically require the Reverse DNS functionality.*

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	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 3.1 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 update (To install Aventail Connect 3.1, you must upgrade Windows 95 with the Microsoft WinSock 2 update prior to Aventail Connect installation and setup. If you do not install the Microsoft patch, Aventail Connect 2.6 will be installed. For more information, see "What Does Aventail Connect Do?".)

Aventail Connect 2.6 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 update (If you do not upgrade Windows 95 with the Microsoft WinSock 2 update, Aventail Connect 2.6 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 <b>Aventail Connect</b> icon in system tray	Double-click <b>Extranet Neighborhood</b> icon on desktop	In system tray	Not available
Windows 3.1, Windows for Workgroups 3.11, Windows NT 3.51	<b>Aventail Connect</b> icon in Aventail Connect program group window	Click <b>Aventail Connect</b> 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 *as31s.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.

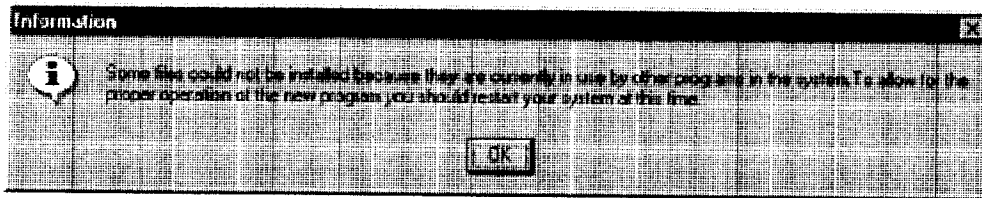
## INSTALLING AVENTAIL CONNECT

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 (Aventail VPN Client or Aventail 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 workstations. 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.



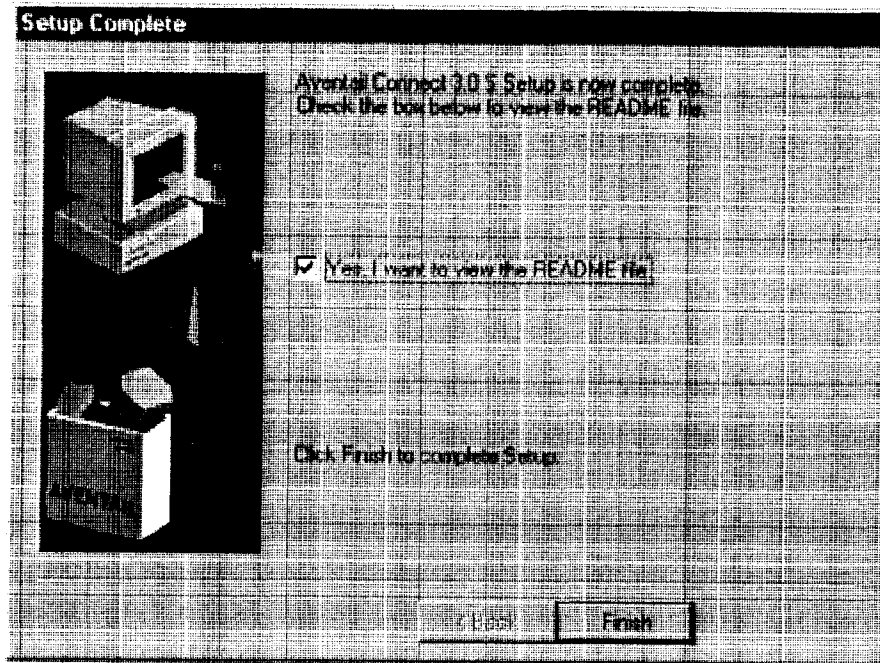
**NOTE:** *You will be asked during the installation procedure if you would like Aventail Connect to be run automatically during startup. In most cases, you will select **yes**. Exceptions to this can be determined by the network administrator.*

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

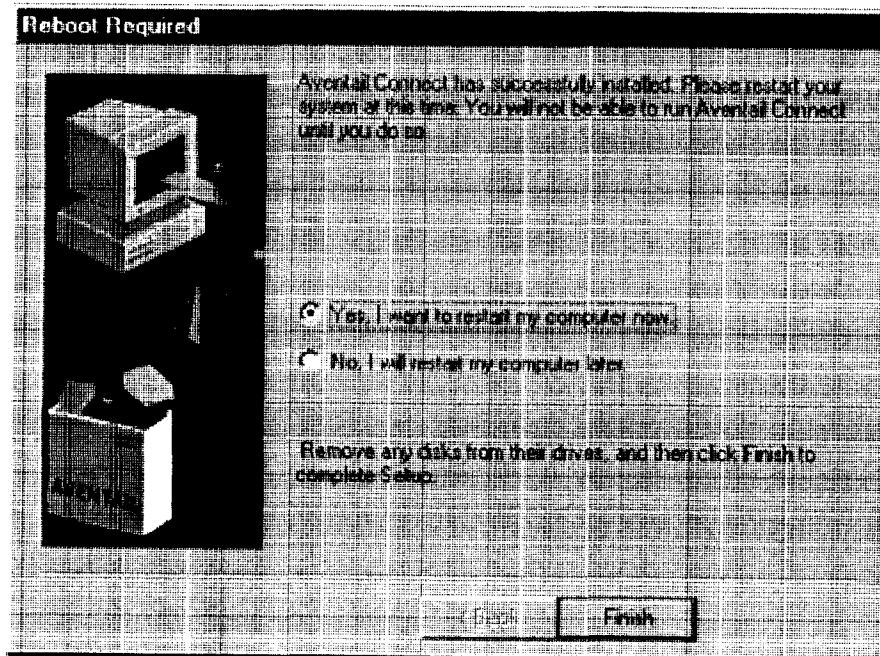
-OR-

Simply click **Finish** in the **Setup Complete** dialog box 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 selected **Yes** when asked if Aventail Connect should be added to your startup directory. (If, during installation, you specified that Aventail Connect *not* be added to the startup directory, 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 then 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 `as31s.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. (For more information, see "Customizer.") 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:

- **Remote UNC:** Remote client configuration file on a Windows share using UNC path and filename (e.g., \\internal\common\a.cfg)
- **Local Configuration File:** Local client configuration file common for all users, but distributed via a locally stored Aventail Connect package
- **Remote Web Server:** Remote configuration files stored on a Web server using URL (e.g., http://internal/a.cfg)

Configuration file setup method	Location	Advantages	Disadvantages
Remote UNC	Windows share using UNC path and filename	<ul style="list-style-type: none"> <li>• Configuration file can be centrally maintained.</li> <li>• No local caching required.</li> </ul>	<ul style="list-style-type: none"> <li>• File server must be on local network. If file server is unavailable, Aventail Connect will not function.</li> </ul>
Local Configuration File	Locally stored setup package	<ul style="list-style-type: none"> <li>• Does not require network connection; configuration file is always available.</li> </ul>	<ul style="list-style-type: none"> <li>• Configuration files cannot be centrally maintained.</li> </ul>
Remote Web Server	Web server	<ul style="list-style-type: none"> <li>• Configuration file can be centrally maintained.</li> <li>• Connection to Web server can be made across the Internet, and can traverse proxies.</li> <li>• Supports authentication and encryption.</li> <li>• If Web server is unavailable, locally cached copy can be used.</li> </ul>	<ul style="list-style-type: none"> <li>• Requires Web server.</li> <li>• Requires network connection for updates.</li> </ul>

### 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 or a Web server. Aventail recommends that you test all configuration files before distribution.

You can 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. For more information, refer to the "Customizer" section.

### To distribute a shared configuration file

There are three methods for distributing shared configuration files.

- **Remote UNC:** Copy the file to a Microsoft or Novell network drive accessible by all users, or 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 UNC's.) If you copy the file to a network drive, 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-

- **Local Configuration File:** Create a shared configuration file to be installed on workstations during the standard Aventail Connect installation/upgrade process. 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.

-OR-

- **Web Server:** Copy the file to a Web server. The Web server can be directly accessible to the workstation, or it can be behind a proxy server. To keep configuration files secure, you can redirect the configuration file connection, authenticated and encrypted, across firewalls.

### **Storing Remote Configuration Files on a Web Server**

When you specify the remote configuration file in Aventail Connect, include the entire URL (e.g., <http://aventail.com/server1/config.cfg>). You can specify this URL in the **Aventail Connect Configuration File** dialog box, or with Customizer.

Aventail Connect keeps a temporary local copy of the remote configuration file in its program directory, with the filename `_ashttpX.cfg`, where X is a number between 0 and 9. Keeping a local copy of the remote configuration file allows the connection to the Web server to be proxied (with authentication and encryption) if necessary. Whenever the remote configuration file needs to be downloaded, Aventail Connect will check the cached copy of the configuration file to determine whether redirection is necessary.

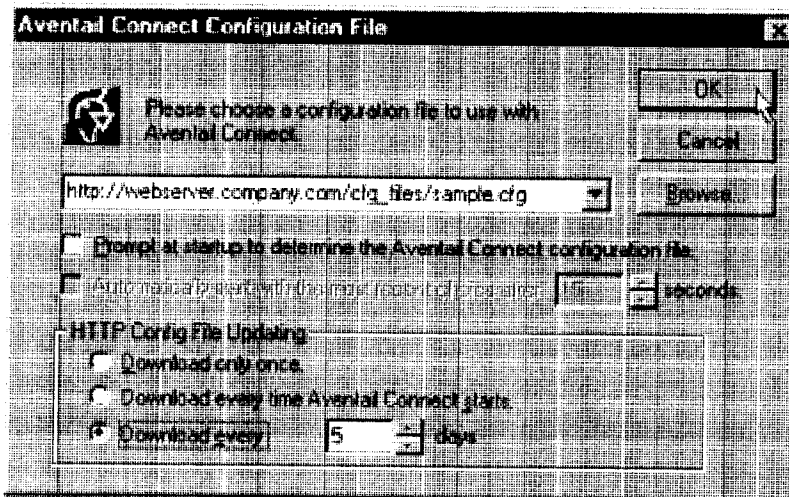
Aventail Connect can download remote configuration files either every time Aventail Connect starts or on a scheduled basis. You can configure this setting in the **Aventail Connect Configuration File** dialog box, or when adding a remote configuration file to a custom installation package with Customizer. When you add a remote configuration file with Customizer, a cached copy of the file can automatically be added to the package.

#### To store remote configuration files on a Web server

1. Place an Aventail Connect configuration file on a Web server.
2. If redirection through a proxy server is required to reach the Web server, configure Aventail Connect to use a configuration file that can access the Web server. If redirection is not required, skip this step.
3. With Aventail Connect running, select **Configuration File** from the system tray menu.

The **Aventail Connect Configuration File** dialog box will open.

4. Enter the URL and filename of the configuration file, e.g., `http://web-server.company.com/cfg_files/sample.cfg`. Click **OK**.



5. Under "HTTP Config File Updating," specify how often Aventail Connect will download the configuration file. Click **OK**.

The configuration file will automatically be downloaded, and Aventail Connect will begin using it immediately. A local copy of the configuration file will be cached in the Aventail Connect program directory.

## 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 can 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. (For more information, see "Network Installation.")

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 (3.1 and 2.6). 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.

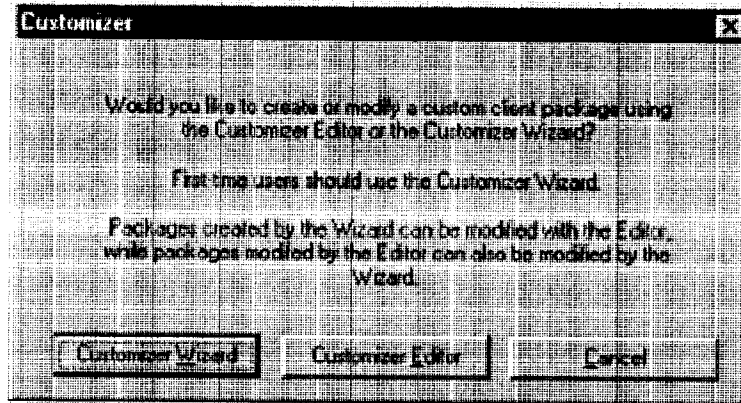
## RUNNING CUSTOMIZER

The Customizer and the Aventail Connect installation package are included in the Customizer directory on the Aventail Connect CD. Before running Custom-

izer, 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.

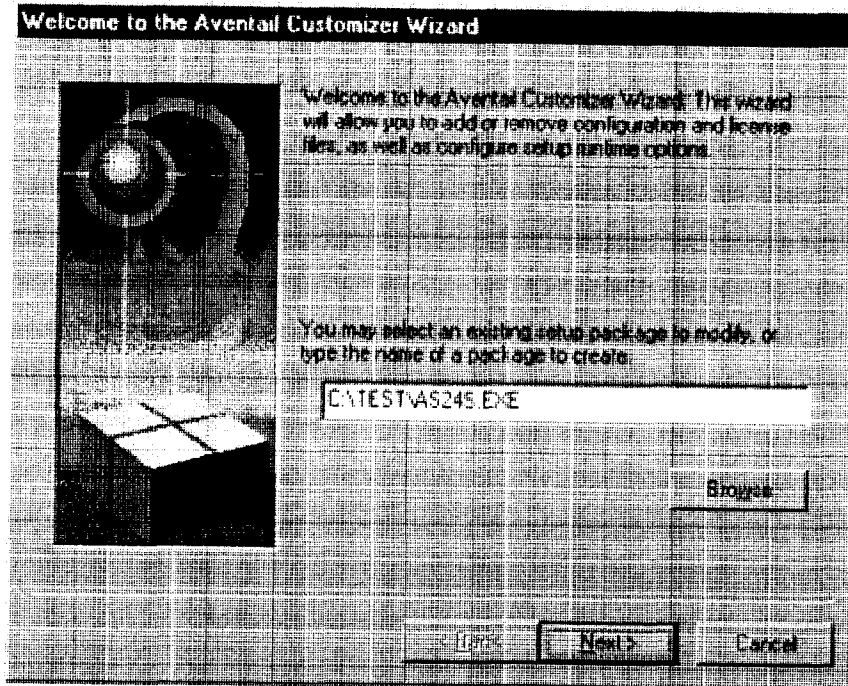
When you run Customizer, you will be prompted to select either the Customizer Wizard or the Customizer Editor.



- **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:

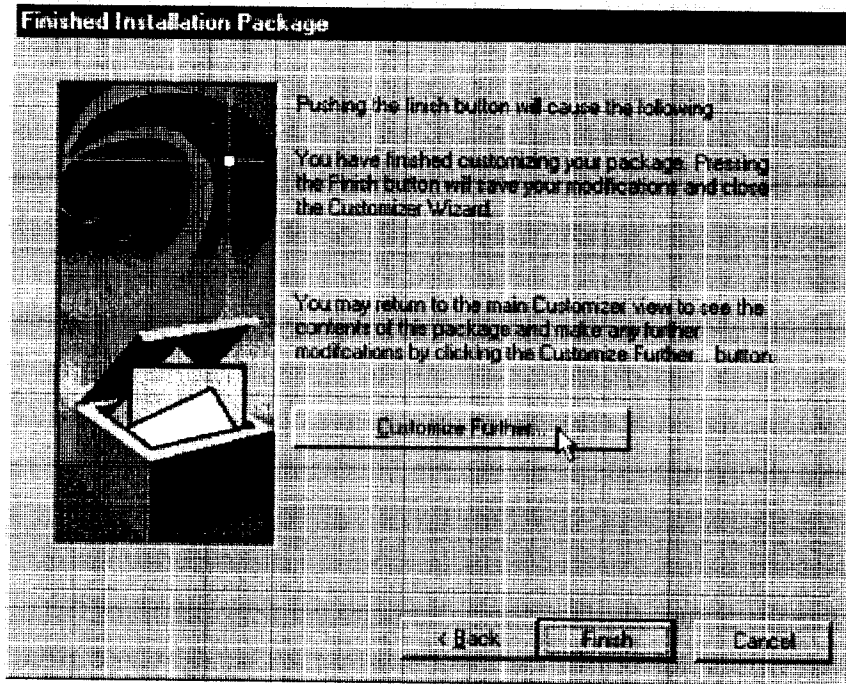
- Specify which platform(s) to support
- Add a license file, or leave an existing license file in the package
- Add or remove configuration files
- Select X.509 certificate files
- Select an extranet hosts (SEHosts) file
- Specify a custom destination directory
- Specify whether or not to put program 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.

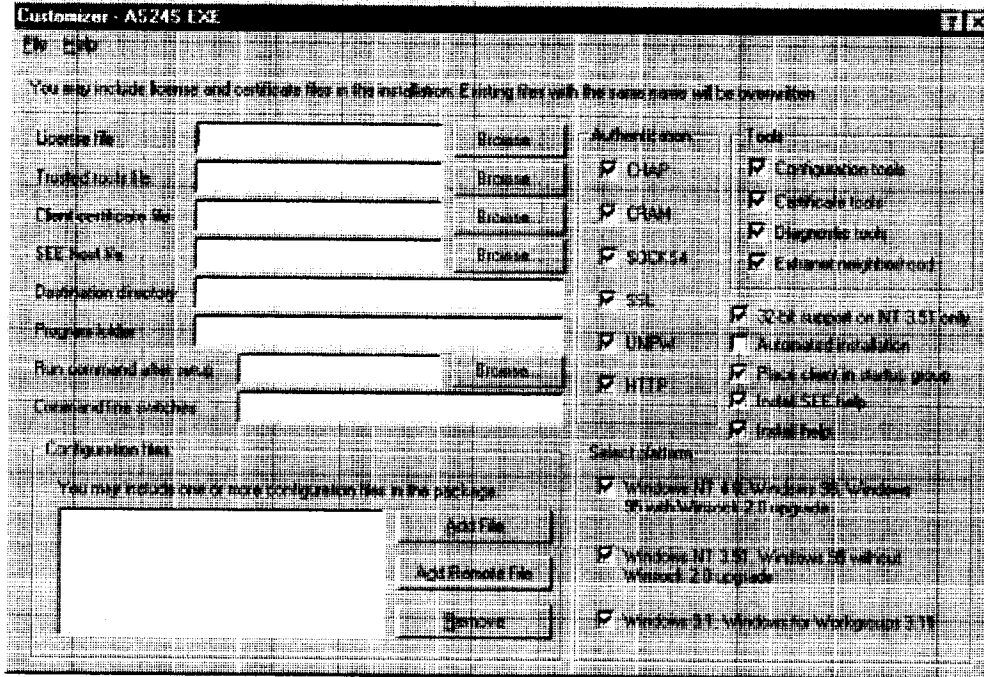
After entering or modifying the package information, the **Finished Installation Package** dialog box appears.



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 <code>aventail.alf</code> )	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 upgrade, Windows 95 without WinSock 2 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 3.1 supports Windows 95 (with the Microsoft WinSock 2 update), Windows 98, and Windows NT 4.0 (with Service Pack 3 or above, available from Microsoft). Aventail Connect 2.6 supports Windows 95 (without the Microsoft WinSock 2 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 package. You must use the packaged Aventail license file, `aventail.alf`.



**CAUTION:** *Aventail Connect 3.1 and 2.6 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 (v2.42 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 2.6 on Windows 95:** By default, Windows 95 does not support WinSock 2, but you can upgrade it to support WinSock 2 with a Microsoft patch. (The patch, `w95ws2setup.exe`, is available from Microsoft, at [http://www.microsoft.com/Windows95/downloads/contents/wuadmin/tools/s\\_wunetworkingtools/W95Sockets2/default.asp](http://www.microsoft.com/Windows95/downloads/contents/wuadmin/tools/s_wunetworkingtools/W95Sockets2/default.asp). However, this procedure adds an extra step to the installation and setup process. Unless users need the MultiProxy feature, which is available only in Aventail Connect 3.1, Aventail recommends that you install Aventail Connect 2.6 rather than 3.1 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 3.1/2.6 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 (v2.42 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 SOCKS servers
2. Define the destinations (networks and hosts)
3. Specify redirection rules
4. Enter 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-

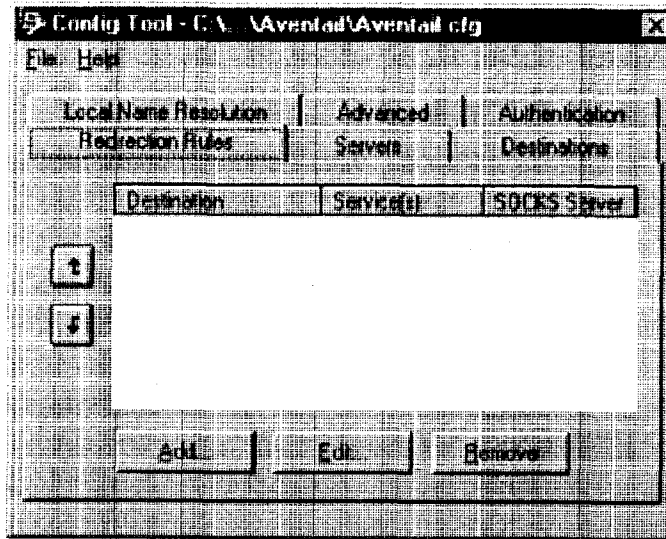
Right-click the **Aventail Connect** icon in the taskbar and click **Config Tool** (Windows 95, Windows 98, or Windows NT 4.0 programs menu option), or double-click the **Config Tool** icon in the Aventail Connect program group (Windows 3.1, Windows for Workgroups 3.11, or Windows NT 3.51).

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).
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.

Aventail Connect 3.1 allows you to create or modify a configuration file and then immediately use it, without needing to restart Aventail Connect and any Aventail-processed applications. When you modify a configuration file, Aventail Connect can re-read the updated configuration file; all applications being processed by

Aventail Connect will then immediately begin using the new configuration information.

When you make a modified configuration file active, Aventail Connect will save the current (modified) configuration file, update the registry, and load the selected configuration file. Aventail Connect will begin using the modified configuration file with any subsequent TCP connection requests, and/or any subsequent UDP activity.



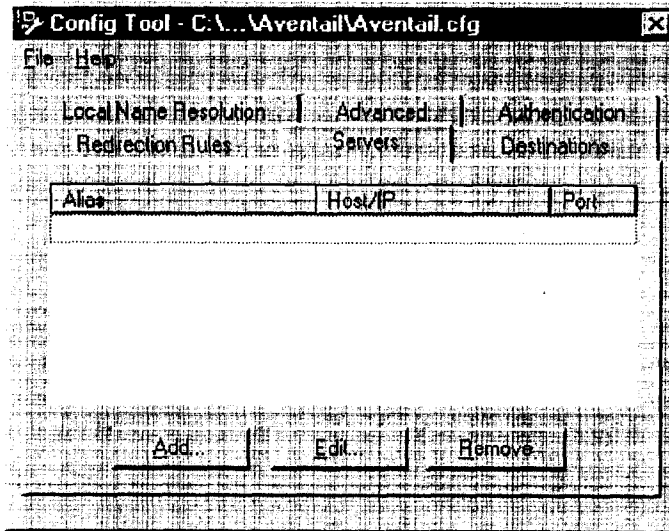
**NOTE:** The configuration file "refresh" feature is supported in Aventail Connect 3.1 only. It is not supported in Aventail Connect 2.6. To activate modified configuration files in Aventail Connect 2.6, you must first shut down and restart Aventail Connect and all applications being processed through Aventail Connect.

**To load a modified configuration file for immediate use**

- With the newly modified configuration file open, select **Make Active** from the File menu of the Config Tool
- OR-
- From the system tray menu, select **Configuration File**, and select (or enter the name of) the configuration file that you want to use. Click **OK**.

**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.

Aventail Connect 3.1 allows you to set a server fallback timeout for every Aventail ExtraNet Server. If a primary SOCKS server is down, or otherwise unable to accept connections, Aventail Connect can fall back to a secondary server. You can set the server fallback timeout, in seconds, on a server-by-server basis. If you do set a server fallback timeout, each connection to a primary server must be completed within the specified length of time or else the connection will fall back to the secondary server.



**NOTE:** *Server fallback timeouts are supported in Aventail Connect 3.1 only. You cannot set a server fallback timeout in Aventail Connect 2.6; you must let the TCP/IP stack time out.*



**NOTE:** *Aventail Connect can fall back to only one server. For example, Aventail Connect could fall back from Server A (primary server) to Server B (secondary server). Aventail Connect could not, however, fall back from Server A to Server B to Server C.*

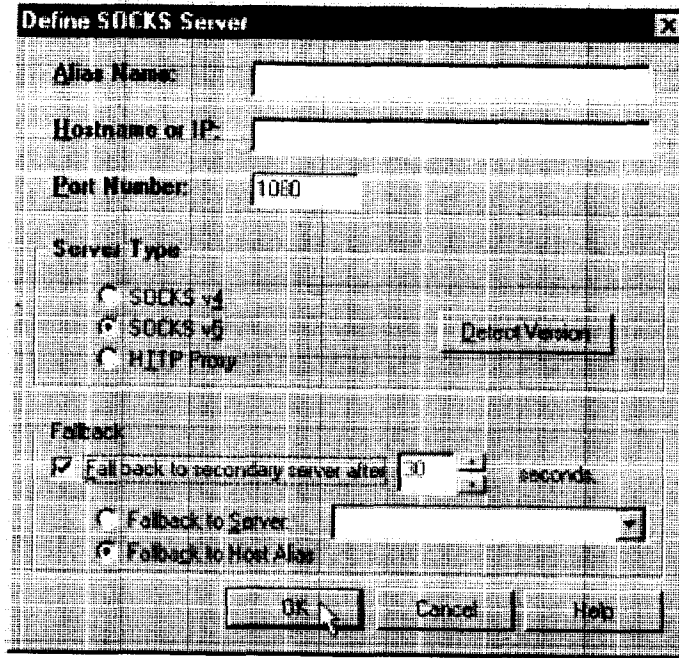
During normal operation, if you configure Aventail Connect to fall back to a secondary server, connections will be directed to the primary server. If the primary server does not respond or accept the connection by the end of the fallback timeout period, the connection will be redirected to the secondary server. If the secondary server accepts the connection, all subsequent connections will automatically be directed to the secondary server. The secondary server is generally meant to be used only when the primary server is unable to accept connections. To prevent the secondary server from automatically becoming the default server for all subsequent connection, Aventail Connect will check the primary server's status every ten minutes. If the primary server is back up and able to accept connection, all subsequent connections will be routed through the primary server.



**CAUTION:** *Do not enable the server fallback option if you are using plug gateways.*

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	Fall back to secondary server after x seconds	Server fallback timeout period (in seconds).
	Fall back to Server:	SOCKS server alias for redundant server.
	Fall back 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. If you want to use a fallback server, select **Fall back to secondary server after...** under "Fallback." Either select **Fall back to server** and directly specify an extranet server for redundancy, or select **Fall back to host alias**. Select or enter, in seconds, the fallback timeout period. Click **OK**.

#### **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, and then click **OK**.

#### **To remove an extranet (SOCKS) server definition**

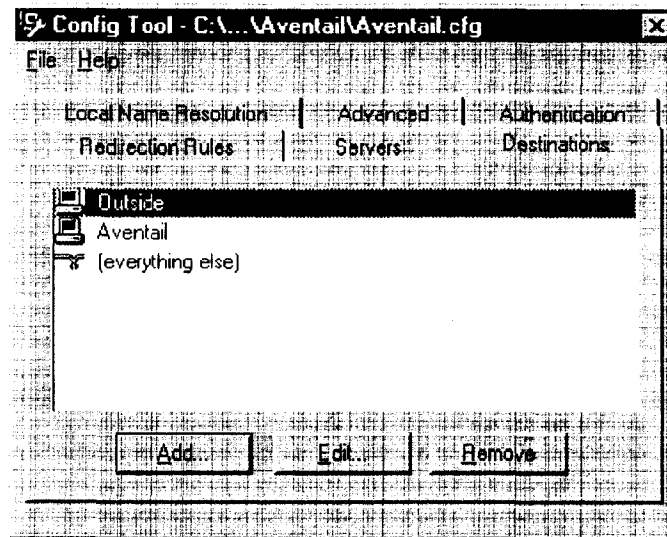
---

- 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 SOCKS 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)."

## WILDCARDS IN HOSTNAME DEFINITIONS

Aventail Connect supports the use of wildcard characters in destination hostnames. You can use wildcards when defining named destinations (hostnames); you cannot use wildcards when defining numerical destinations, such as IP addresses or subnet masks.

Acceptable wildcard characters are "?" and "\*" (where "?" represents one character, and "\*" represents any number of characters). For example:

```
e*tra.in.aventail.com matches extra.in.aventail.com
e?tra.in.aventail.com matches extra.in.aventail.com
e?ra.in.aventail.com does NOT match extra.in.aventail.com
```

You can use any combination of "?" and "\*" characters between each set of periods. However, each section must contain at least one non-wildcard character. For example, the following destination names would be allowed:

```
e?t?a.in.aventail.com
*xtr?.in.aventail.com
e???a.in.ave*.com
e*.in.*tail.com
```

The following destination names, however, would not be allowed:

extra.\*.aventail.com  
 \*.\*.aventail.com  
 extra.in.\*.com



**CAUTION:** You cannot use a wildcard character, or a series of wildcard characters, to represent multiple sections. Any wildcard character in a section can represent characters within that section only. For example:

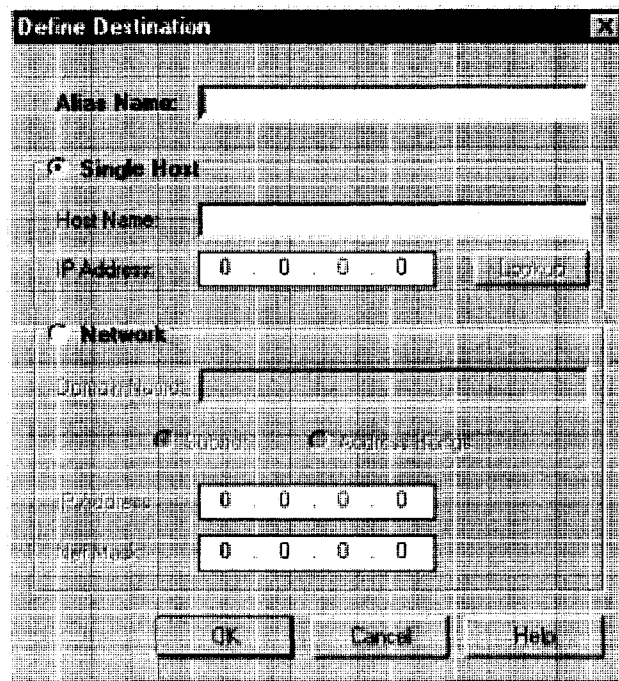
e\*.in.aventail.com **matches** extra.in.aventail.com  
 e\*.aventail.com **does NOT match** extra.in.aventail.com

**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 (optional)	Full numeric IP address
	Lookup	Look up IP address
Network	One or more computers in a network	
	Domain Name	Domain of the network
	Subnet (optional)	IP address and netmask address
	Address Range (optional)	Beginning and ending IP addresses From Starting IP address To Ending IP address



**CAUTION:** *The IP Address, Subnet, and Address Range fields are all optional. However, in order to apply redirection rules when connecting by IP address, you must enter IP address and subnet information.*

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, if applicable, 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**

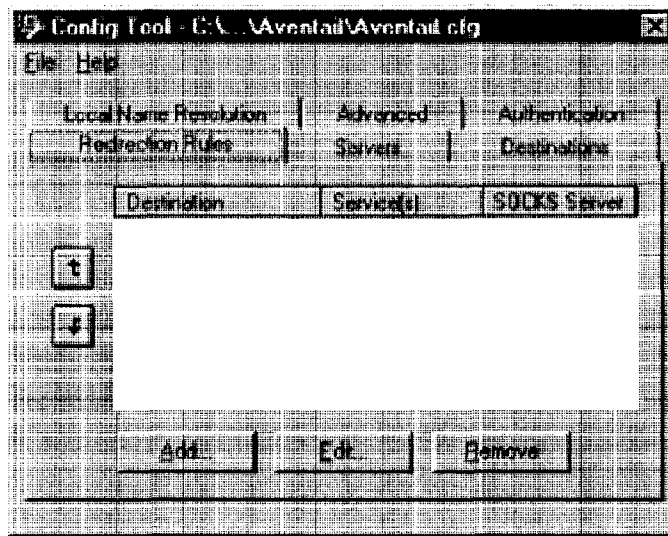
- Select the destination you want to remove and click **Remove**.

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 <b>Destinations</b> 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**

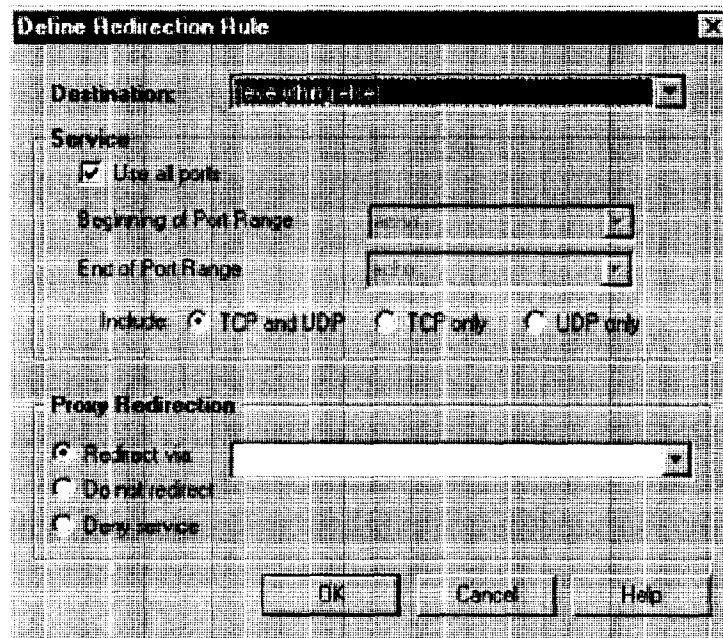
As you add destinations, use the arrow buttons to prioritize them. List the most specific rules first and the general rules last.



**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.

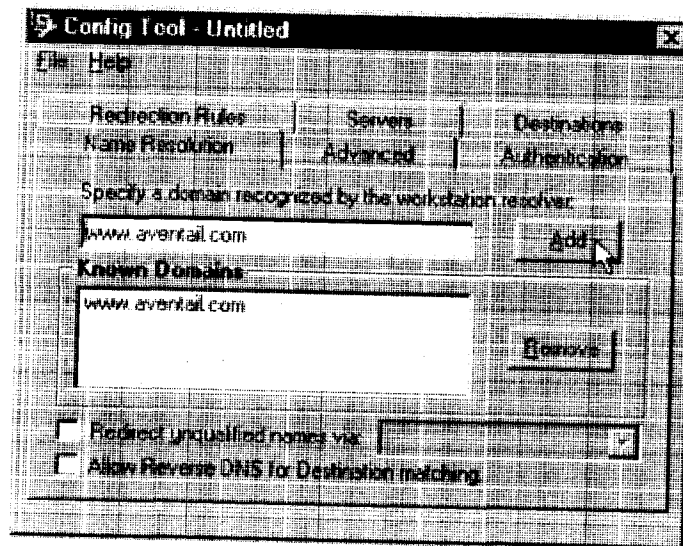
## DEFINE NAME RESOLUTION

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 **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.

Name Resolution is specified on the **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
Allow Reverse DNS for destination matching	Enable Reverse DNS (converts IP addresses into hostnames)

**To add a local domain name**

- On the **Name Resolution** tab, type the new name in the **Specify a domain** box and click **Add....**
- If necessary, select **Allow Reverse DNS for destination matching**.  
The new name is moved into the **Known Domains** box. It is now active.



**CAUTION:** *The reverse DNS process can create unexpected delays, causing Aventail Connect to behave unpredictably. Aventail recommends that you do not enable this option unless you specifically require the Reverse DNS functionality.*

**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.*

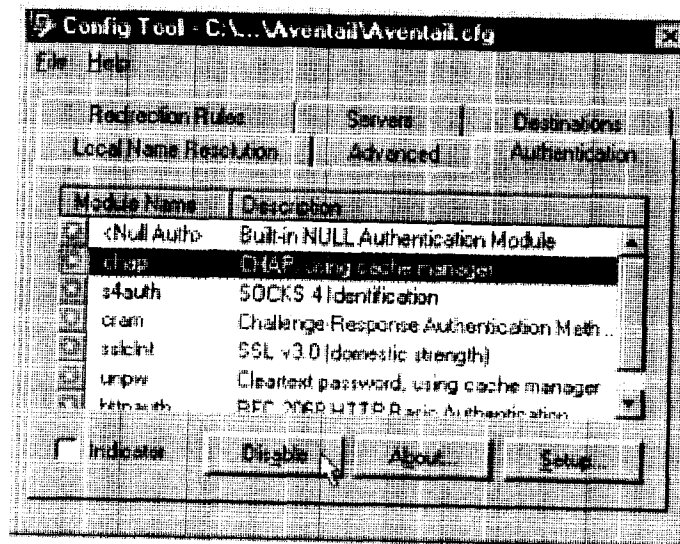
The current Aventail Connect authentication modules are SOCKS v4 Identification, Username/Password, Challenge Handshake Authentication Protocol (CHAP), Challenge Response Authentication Method (CRAM), Secure Sockets Layer (SSL), and HTTP Basic (username/password). Each of these authentication modules 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).

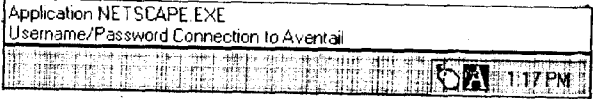
Aventail Connect can cache authentication credentials in memory, based on the option you select in the **Authentication** dialog box. Memory caching stores the credentials for the current session only. When you restart Aventail Connect or Windows, the memory cache is flushed and you must reenter your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

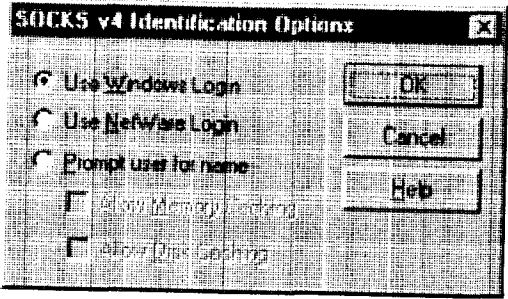
Aventail Connect includes backward compatibility for the SOCKS 4 protocol. SOCKS 4 does not support password authentication, so only your username is sent unencrypted to the extranet (SOCKS) server along with your connection request.

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.				
	<table border="1"> <tr> <td>Allow Memory Caching</td> <td>Stores credentials in memory for this session only. Cache is flushed upon restart; credentials must be reentered as prompted.</td> </tr> <tr> <td>Allow Disk Caching</td> <td>This option is currently unavailable. (Stores credentials on disk for future sessions.)</td> </tr> </table>	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.)
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.)				

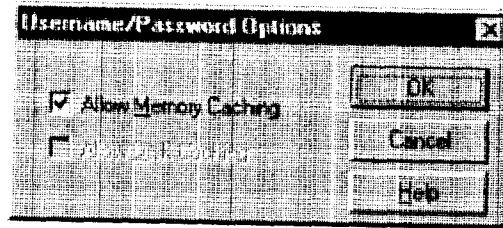
- When you select the **Prompt user for name** option, you must also select the desired caching option. (Currently only Memory Caching is available.)
- After making appropriate selections, click **OK**.

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) user-name 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.

- 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**.

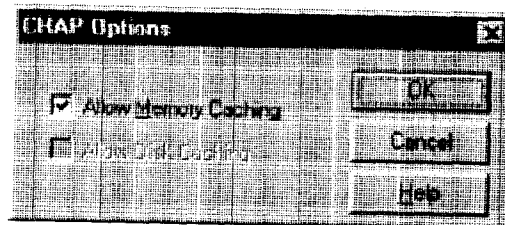
The dialog box closes and the Config Tool reappears.

**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.



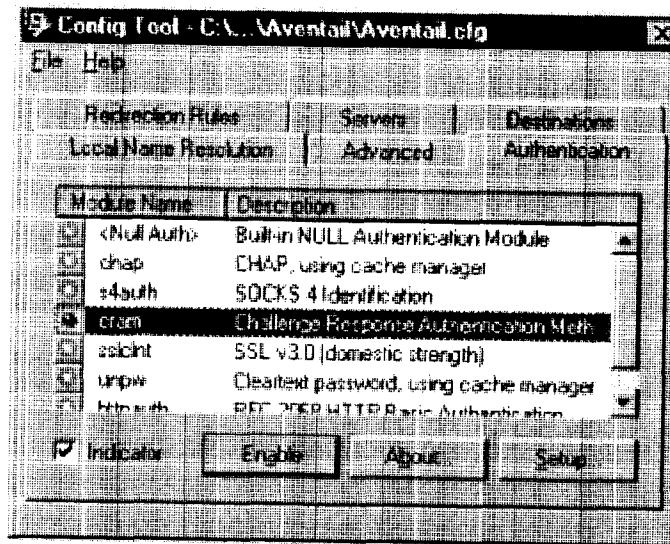
Field	Description
Allow memory caching	Stores credentials in memory for this session only. Cache is flushed upon restart; credentials must be reentered as prompted.
Allow disk caching	This option is currently unavailable. (Stores encrypted credentials on disk for future sessions.)

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) v3.0, 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.

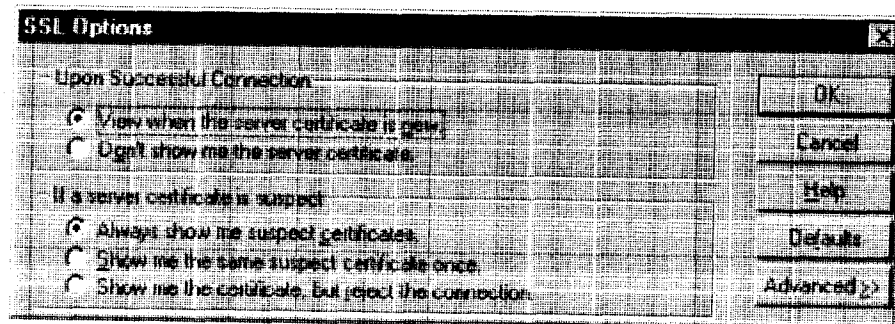
Certificates contain special integrity checks and electronic signatures that verify that the certificate is genuine, was issued by a certification authority, and was not tampered with. Anybody can issue a certificate that says anything; the client must know who issued the certificate, and have some trust 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 **sslCInt** (SSL v3.0) 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:

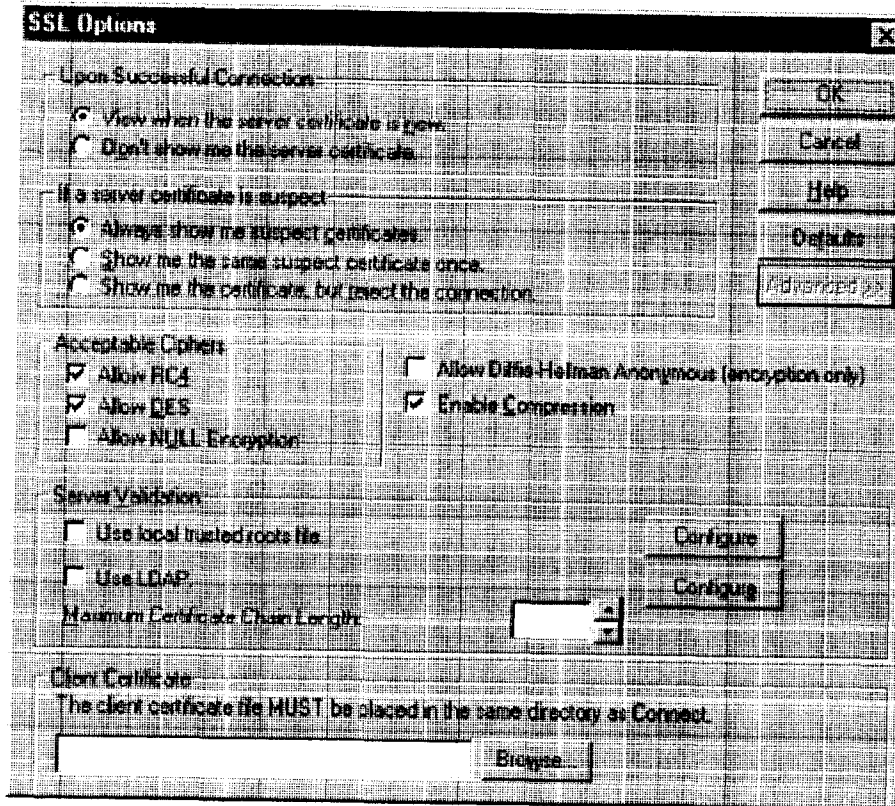
- **Always show me suspect certificates:** Aventail Connect will display suspect certificates each time they are received. The **Certificate** dialog box will appear for each new connection to the server(s) sending a suspect certificate. (This option allows you to continue the connection despite the 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:** Aventail Connect 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:** Aventail Connect will reject a connection if the certificate is suspect. It will display the certificate to allow you to view it.
4. Click **Advanced** in the dialog box to show the acceptable cipher (a cryptographic algorithm used to encrypt the data stream) options.



Field	Description
Acceptable Ciphers	
Allow RC4	Offer the RC4 cipher to the server.
Allow DES	Offer the DES cipher to the server.
Allow NULL Encryption	Do not encrypt using SSL. SSL will be used to authenticate only.
Allow Diffie-Hellman Anonymous	Do not authenticate the server; only do encryption.
Enable Compression	Use SSL compression to improve performance when slower connections are detected.
Server Validation	
Trusted Roots	Use a trusted roots file to validate trusted certificate chain roots. <i>NOTE: The trusted roots file MUST be placed in the same directory as the Aventail Connect configuration file</i>
	Configure      Configure trusted roots
LDAP	Use an LDAP server to validate trusted certificates.
	Configure      Configure LDAP
Maximum Chain Length	Specify the maximum allowable certificate-chain length.
Client Certificate	Select a client certificate file. <i>NOTE: The client certificate MUST be placed in the same directory that Aventail Connect was installed to.</i>
	Browse      Select the specific file

During the initial SSL connection, the client and the server negotiate which cipher to use. Checking a particular cipher in the dialog box does not mean that it will be used. Instead, each checked cipher is *offered* to the server, but the server determines which cipher to use. If the server requires a cipher that is not selected in this dialog box, the authentication will fail.

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

- **Allow RC4:** Aventail Connect encrypts the information using the RC4 cipher.
- **Allow DES:** Aventail Connect encrypts the information using the DES cipher.
- **Allow NULL Encryption:** Aventail Connect allows the server to select *no* encryption. Message integrity is still assured, but the data will be sent in cleartext.
- **Allow Diffie-Hellman Anonymous:** Aventail Connect will be able to communicate with the extranet (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).

- **Enable Compression:** To speed the encryption process and enhance overall performance, Aventail Connect will automatically compress encryption when a narrow bandwidth and/or slow modem are detected.
5. If necessary, add (or delete) a trusted roots (\*.rot) file and/or an LDAP server definition.

**To add or remove a trusted root**

- a. In the **SSL Options—Advanced** dialog box, under “Server Validation,” select **Use local trusted roots file**, and then click **Configure**.

The **Trusted Roots** dialog box will appear.

- b. Enter the name of the trusted roots file, or click **Browse** to search for the file, and then click **OK**.



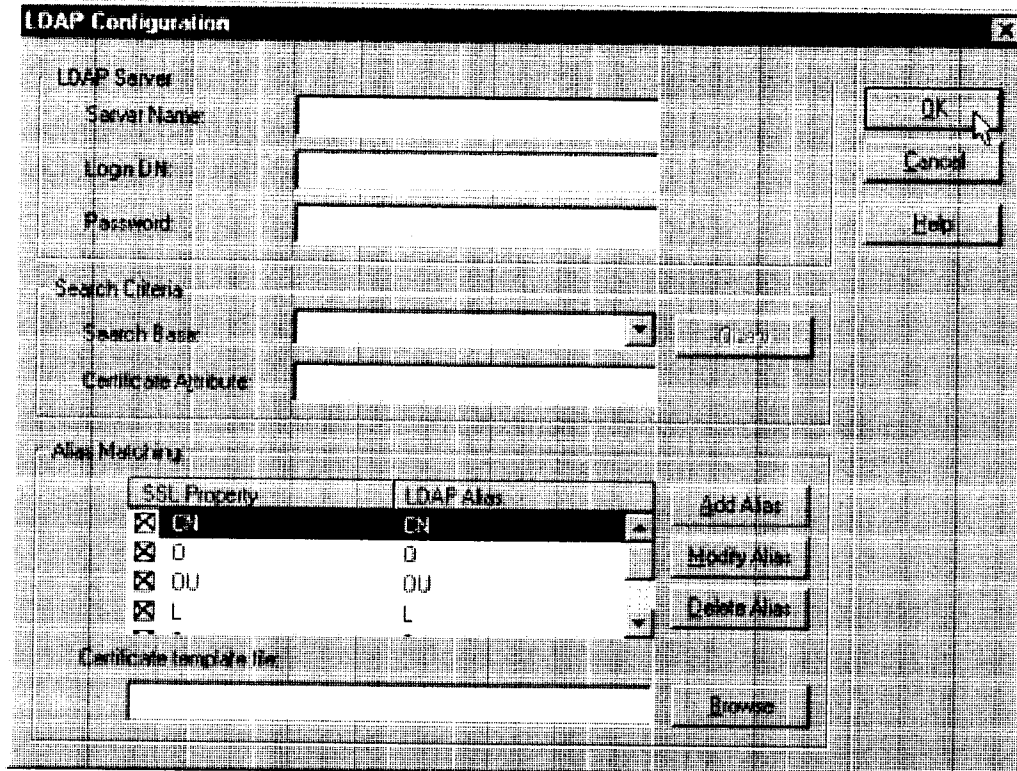
**CAUTION:** *The trusted root file must be in the same directory as the Aventail Connect configuration file.*

**To configure LDAP**

- a. In the **SSL Options—Advanced** dialog box, under “Server Validation,” select **Use LDAP**, and then click **Configure**.

The **LDAP Configuration** dialog box appears.





The image shows a dialog box titled "LDAP Configuration" with a close button in the top right corner. The dialog is divided into several sections:

- LDAP Server:** Contains three text input fields labeled "Server Name", "Login DN", and "Password".
- Search Criteria:** Contains a "Search Base" dropdown menu, a "Certificate Attribute" text input field, and an "Add" button.
- Alias Matching:** Contains a table with two columns: "SSL Property" and "LDAP Alias". The table has four rows, each with a checked checkbox in the first column. To the right of the table are three buttons: "Add Alias", "Modify Alias", and "Delete Alias".
- Certificate template file:** Contains a text input field and a "Browse" button.

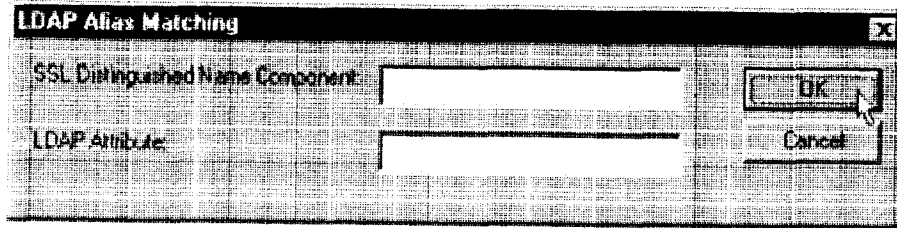
Buttons for "OK", "Cancel", and "Help" are located on the right side of the dialog.

SSL Property	LDAP Alias
<input checked="" type="checkbox"/> CN	CN
<input checked="" type="checkbox"/> O	O
<input checked="" type="checkbox"/> OU	OU
<input checked="" type="checkbox"/> L	L

Field	Description	
LDAP Server		
Server Name	Enter the LDAP server hostname.	
Login DN	Enter the login DN (distinguished name) for the LDAP server.	
Password	Enter the password for the LDAP server.	
Search Criteria		
Search Base	Enter the DN to use as the search base.	
	Query	Search available DN's to use as search base.
Certificate Attribute	Enter the certificate attribute.	
Alias Matching		
SSL Property/LDAP Alias	Names of SSL property and corresponding LDAP alias.	
Add Alias	Add an LDAP alias/SSL property.	
Modify Alias	Modify an LDAP alias.	
Delete Alias	Delete an LDAP alias/SSL property.	
Certificate template file:	(Optional) Enter name of certificate file to use as template.	
	Browse	Search available certificate files.

- b. Under "LDAP Server," enter the LDAP server name, and the DN and password that you want to log in under.
- c. Under "Search Criteria," enter or select the DN to use as the search base, and enter the certificate attribute. (In most cases, the certificate attribute will be "usercertificate.")
- d. Under "Alias Matching," select the SSL properties that you want to use as search criteria.

If necessary, you can modify any of the LDAP aliases to map to the SSL properties. To modify an LDAP alias, click **Modify Alias**. In the **LDAP Alias Matching** dialog box, enter the LDAP Attribute that will map to the SSL Distinguished Name Component. You can also **Add** or **Remove** an SSL property/LDAP alias in the **LDAP Alias Matching** dialog box.



In the **Certificate template file:** box, you can specify a certificate file to use as a template. If you specify a certificate template file, Aventail Connect will automatically populate the "SSL Property/LDAP Alias" box with the attributes used in the specified certificate template file.

- e. Click **OK**.
- 6. If Aventail Connect sends a client certificate to the server during the initial authentication exchange, it sends the certificate identified in the **Client Certificate** window. To load the client certificate, press **Browse** and then select the client certificate (\*.cer) from the Aventail Connect directory. Only the file-name of the certificate file loads via the **Browse** button, and not the path-name.



**CAUTION:** *The client certificate file must be placed in the Aventail Connect directory.*

When Aventail Connect receives a certificate from a server, it looks at the root of the certificate chain and matches it against the Aventail Connect list of trusted roots.

You can specify the maximum number of certificates in a certificate chain. The default maximum length is two certificates. In most instances, Aventail recommends allowing no more than two certificates to form a chain, although you can specify up to ten. The longer the certificate chain, the less secure the chain is.



**CAUTION:** *In most instances, Aventail recommends allowing no more than two certificates in a certificate chain. Allowing more than two certificates can compromise security.*

- 7. After making appropriate selections, click **OK**.

### PKCS #12 CERTIFICATES FOR USER AUTHENTICATION

Aventail Connect supports PKCS #12-formatted X.509 client certificates for SSL authentication. PKCS #12-formatted certificates are stored in a portable format for easy exchange between applications. You can generate client certificates by enrolling with a public-key infrastructure (PKI), such as VeriSign OnSite. You can then use your Web browser to export the client certificate to a PKCS #12 file in

the Aventail program directory. When users connect to an Aventail ExtraNet Server for the first time, they will be prompted to select a certificate.

**To export a PKCS #12-formatted X.509 certificate**

1. Using a Web browser and a CA, such as VeriSign Onsite, obtain a client certificate.
2. Export the certificate to a file in the Aventail program directory. You can use any filename. This step varies from browser to browser.

**Microsoft Internet Explorer 4.01**

- a. Select **View|Internet Options...|Content|Certificates|Personal...**
- b. Select the certificate that you want to export, and click **Export...**
- c. Specify a password to protect the certificate.
- d. Save the file to the Aventail Connect program directory.



**CAUTION:** *On Windows NT, Microsoft Internet Explorer 4.01 does not export PKCS #12 certificates properly. This problem was corrected in Microsoft Internet Explorer 5.0.*

**Microsoft Internet Explorer 5.0**

- a. Select **Tools|Internet Options...|Content|Certificates|Personal...**
- b. Select the certificate that you want to export, and click **Export...**
- c. In the Certificate Export Wizard, click **Export the Private Key**.
- d. Specify a password to protect the certificate.
- e. Select the PKCS #12 format.
- f. Select **Include all certificates in the certificate path if possible**.
- g. Save the file to the Aventail Connect program directory.

**Netscape Navigator 4.5**

- a. Click the Lock icon in the lower-left corner of the main Netscape Navigator window.
  - b. Select **Certificate|Yours**.
  - c. Select the certificate that you want to export, and click **Export**.
  - d. Specify a password to protect the certificate.
  - e. Save the file to the Aventail Connect program directory.
3. Use an Aventail Connect configuration file and server setup that forces the user to authenticate using client certificates. Configure the Aventail ExtraNet Server.
  4. Initiate a connection that forces the user to authenticate. You will be prompted for a certificate file. Select the certificate that you just exported, and then click **OK**.

## PKCS #11 SMART CARDS FOR USER AUTHENTICATION

Aventail Connect can use client certificates that are stored on PKCS #11-compatible smart cards for SSL authentication. Currently, Aventail Connect supports the DataKey and SpyruS Rosetta smart cards.

Aventail Connect will be prompted for a file (or smart card) containing certificate information only when the SOCKS server requests client authentication using a certificate. If a SOCKS server requests client authentication with a certificate, and no certificate is already specified for that host, the user will be prompted to select a certificate. You can configure passwords or PINs to be cached to memory, or you can specify that users enter passwords or PINs each time they use a smart card to authenticate.

### To configure PKCS #11 smart-card user authentication

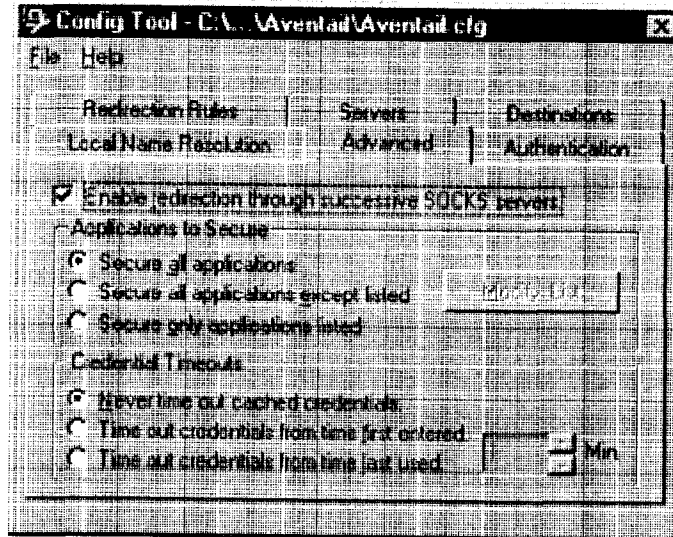
1. Use a smart card with an X.509 certificate stored on it.
2. Install the appropriate smart card software on the user's computer.
3. Include the public certificates of the CA (and any intermediary CAs) for the client certificate in the trusted roots file that Aventail Connect is configured to use.
4. Configure Aventail Connect to redirect to an Aventail ExtraNet Server that requires client certificates.
5. Initiate a connection.
6. When prompted, specify whether you want to authenticate with a client certificate that is stored in a file, a client certificate that is stored on a smart card, or no client certificate at all.
7. Aventail Connect will prompt you for the path of the dynamic link library (DLL) for the smart card's PKCS #11 module. This is the same DLL that is used with Netscape Navigator. Enter the DLL pathname and click **OK**.
8. Aventail Connect will display a list of all detected smart cards on the system. If you have not yet inserted your smart card into the appropriate reader, insert it and click **Refresh List**.
9. Select your smart card and click **OK**.
10. If the smart card is protected with a PIN, you will be prompted to enter it.
11. Select the private key you want to use, and click **OK**.



**NOTE:** Once you specify a smart card token or client certificate to be used with a server, this setting will be remembered indefinitely. To reset the setting, select **Credentials** from the Aventail Connect system tray menu, select (highlight) the credentials, and click **Delete**. Your PIN will not be remembered.

## ADVANCED TAB OPTIONS

The **Advanced** tab in the Config Tool contains three advanced options. In the **Advanced** tab, you can allow SOCKS tunneling through successive extranet (SOCKS) servers, secure selected applications, and set credential cache time-outs.



### ALLOW SOCKS TUNNELING THROUGH SUCCESSIVE EXTRANET SERVERS

Once servers and destinations are defined, you can direct SOCKS traffic through successive extranet (SOCKS) servers.

On the **Advanced** tab in the Config Tool, select the **Enable redirection...** box to allow credential information to forward to successive extranet servers.

### SECURE SELECTED APPLICATIONS

This option allows you to:

- secure all applications except those listed,
- secure only the applications that are listed,
- or secure all applications, enabling neither exclusion nor inclusion.



**NOTE:** You can exclude and include only 32-bit applications. You cannot exclude and include 16-bit applications.

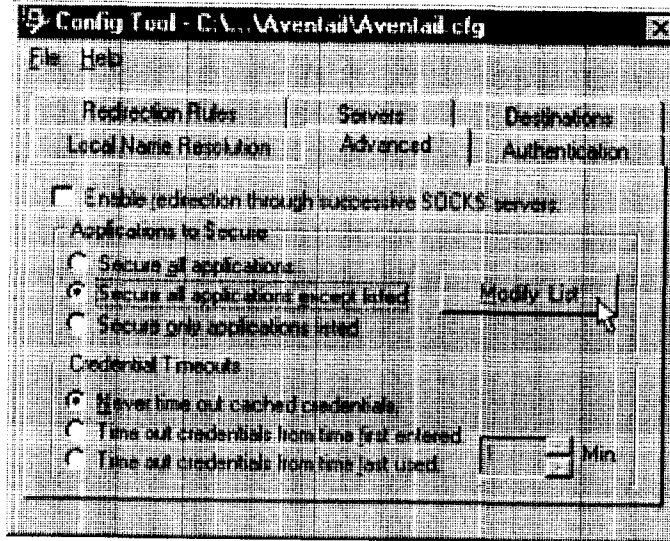
You can exclude or include specified applications in the Exclusion/Inclusion List. With the Exclusion/Inclusion List, you can secure all applications *except* those on the list, or you can secure *only* those applications on the list. The default setting is to secure (hook) *all* network applications.

### Excluding Applications

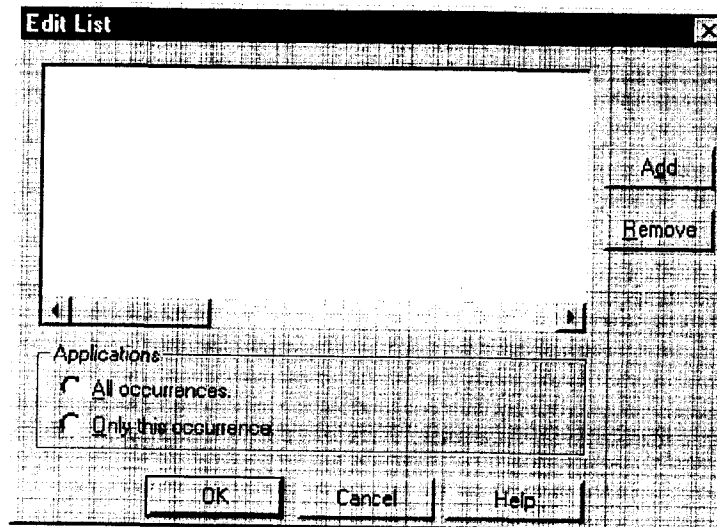
You can exclude specific applications through the Exclusion/Inclusion List. When you enable the "Secure all applications except listed" option, Aventail Connect will not proxy any applications that are on the Exclusion/Inclusion List.

To exclude an application

1. Under "Applications to Secure," select **Secure all applications except listed** and click **Modify List**.

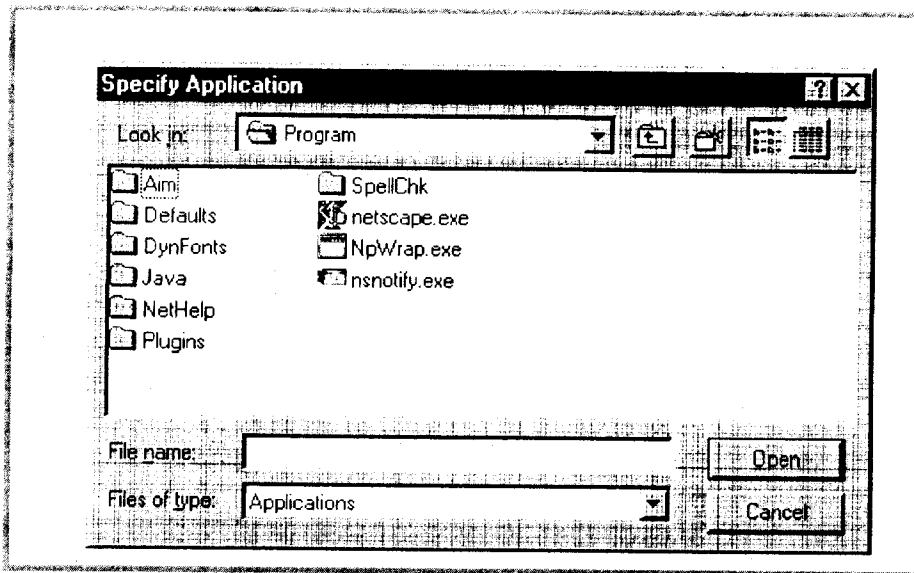


The Edit List dialog box appears.



2. Click **Add**....

The **Specify Application** dialog box appears.



3. Highlight the application(s) to add to the Exclusion/Inclusion List, and then click **Open**.

The **Specify Application** dialog box disappears and the applications are now in the **Edit List** dialog box.

4. In the **Edit List** dialog box, select **All occurrences** or **Only this occurrence**.



**NOTE:** You may have more than one path (instance) of a specified file-name (e.g., ftp.exe). You can choose to exclude one specified application, with a fully qualified pathname (e.g., C:\Windows\Sys32\ftp.exe), or all instances of a specified filename (e.g., all instances of ftp.exe).

- **Only this occurrence:** Selecting this option excludes only the specified application.
- **All occurrences:** Selecting this option excludes all applications with the specified filename.

#### To undo application exclusion

1. Under "Applications to secure," select **Secure all applications except listed**, and then click **Modify List**.

The **Edit List** dialog box appears.

2. Highlight the application you want to remove from the Exclusion/Inclusion List, and then click **Remove**.

The application is removed from the Exclusion/Inclusion List.



### Including Applications

You can include specific applications through the Exclusion/Inclusion List. When you enable the "Secure only applications listed" option, Aventail Connect will hook only those applications that are on the Exclusion/Inclusion List.

#### To include an application

1. Under "Applications to secure," select **Secure only applications listed**, and then click **Modify List**.

The **Edit List** dialog box appears.

2. Click **Add**.

The **Specify Application** dialog box appears.

3. Highlight the application(s) to add to the Exclusion/Inclusion List, and then click **Open**.

The **Specify Application** dialog box disappears and the applications are now in the **Edit List** dialog box.

4. In the **Edit List** dialog box, select **All occurrences** or **Only this occurrence**.



**NOTE:** You may have more than one instance of a specified application (e.g., `ftp.exe`). You can choose to include one specified application, with a fully qualified pathname (e.g., `C:\Windows\Sys32\ftp.exe`), or all instances of a specified application (e.g., all instances of `ftp.exe`).

- **Only this occurrence:** Selecting this option excludes only the specified application.
- **All occurrences:** Selecting this option excludes all applications with the specified filename.

#### To undo application inclusion

1. Under "Applications to secure," select **Secure only applications listed**, and then click **Modify List**.

The **Edit List** dialog box appears.

2. Highlight the application you want to remove from the Exclusion/Inclusion List, and then click **Remove**.

The application is removed from the Exclusion/Inclusion List.

### Securing all Applications

You can secure *all* applications, enabling neither exclusion nor inclusion. When you secure all applications, Aventail Connect ignores any applications on the Exclusion/Inclusion List.

**To secure all applications**

- On the **Advanced** tab, under "Applications to Secure," select **Secure all applications**.



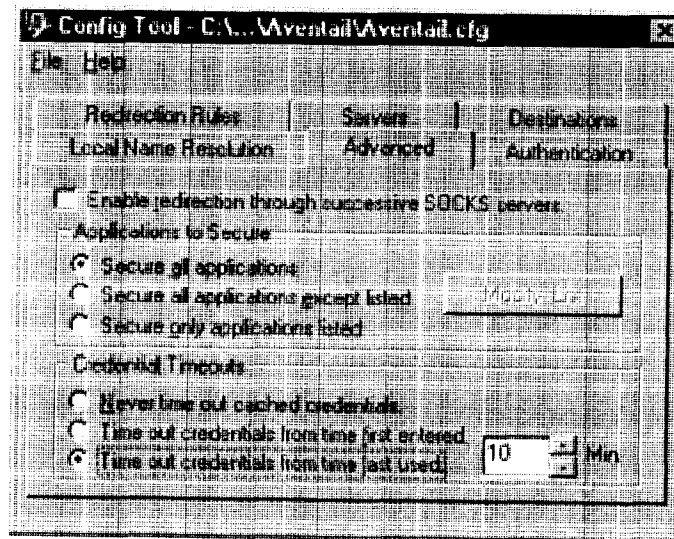
**NOTE:** *Aventail Connect secures all applications by default. Unless you need to exclude or include specific applications, Aventail recommends that you use the default **Secure all applications** setting.*



**CAUTION:** *Microsoft Internet server products (including Microsoft Internet Information Server (IIS) and Microsoft Peer Web Server) include inetinfo.exe, which conflicts with Aventail Connect 3.1. To eliminate this conflict, exclude inetinfo.exe through the Application Exclusion/Inclusion List in the Config Tool.*

**CREDENTIAL CACHE TIMEOUTS**

With the credential cache timeout feature, you can control when credentials expire (time out). If a user has not made a connection to the extranet (SOCKS) server for a certain length of time (determined by the administrator), then the credentials will automatically be deleted from the credential cache. If a credential times out, the user must reauthenticate by entering the proper credentials before regaining access to the extranet. This feature can help to prevent unauthorized users from gaining access to secured areas.



There are three credential cache timeout options.

- **Never time out cached credentials:** Credentials never time out.

- **Time out credentials from time first entered:** Credentials time out *x* minutes after the user first entered the credentials (where "*x*" is the number of minutes you enter in the **Min.** box).
- **Time out credentials from time last used:** Credentials time out *x* minutes after the user last connected through the extranet server (where "*x*" is the number of minutes you enter in the **Min.** box).



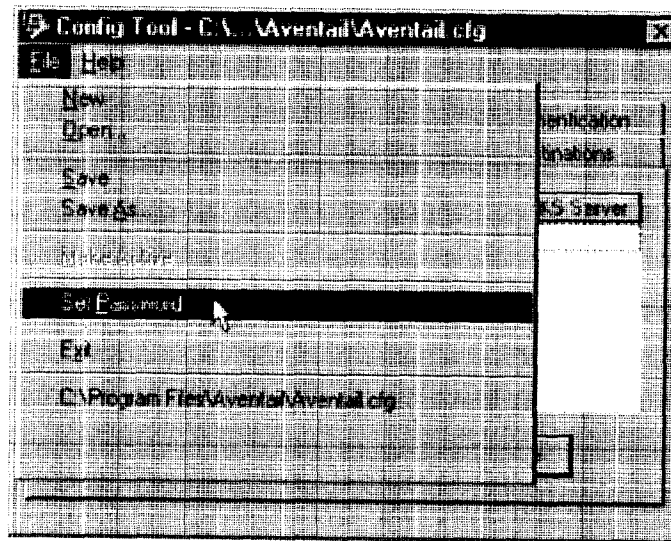
**CAUTION:** *If your mail program is configured to check for e-mail at regular intervals, the mail-checking frequency must be longer than the credential cache timeout. For example, if your mail program is configured to check for mail every ten minutes, you should set the credential cache to less than ten minutes.*

## ENABLE PASSWORD PROTECTION

You can enable password protection for a configuration file. If you enable password protection, users will not be able to view or modify the configuration file without the assigned password. A password is not required to use the configuration file with Aventail Connect.

### To enable password protection

1. From any tab of the Config Tool, select **File | Set Password**.



The **Configuration File Password** dialog box will appear.

2. Enter the desired password.
3. Reenter the password to confirm, and then click **OK**.

### To disable password protection

1. From any tab of the Config Tool, select **File | Set Password**.  
The **Configuration File Password** dialog box will appear.
2. Clear the password from both boxes, and then click **OK**.



**NOTE:** If you save an existing configuration file using the **Save As** command, Aventail Connect will prompt you to enter the correct password for the configuration file.

## MULTIPLE FIREWALL TRAVERSAL

To gain access to your extranet, users may need to traverse multiple firewalls. In the simplest case, this involves an employee at a partner company gaining access to the Internet via an outbound proxy server at the partner company, and having an authenticated, encrypted, and controlled connection to your internal network via an Aventail ExtraNet Server. This capability is provided in Aventail Connect 3.1 by the Aventail MultiProxy feature. Aventail Connect can open connections through SOCKS servers, through HTTP proxies, or through proxy chaining.

- **MultiProxy with SOCKS Server:** Uses a SOCKS server to control outbound access.
- **MultiProxy with HTTP Proxy:** Uses an HTTP proxy to control outbound access.
- **Proxy Chaining:** Uses two Aventail ExtraNet Servers, where one Aventail ExtraNet Server acts as a client to another Aventail ExtraNet Server.

## AVENTAIL MULTIPROXY

The Aventail MultiProxy feature allows Aventail Connect to traverse multiple firewalls by making connections through successive proxy servers. Aventail Connect makes a connection with each proxy server individually. Each proxy server forms a link in a chain that connects Aventail Connect to the final destination. Any or all of the proxy servers can apply authentication and access control rules. Proxies can be Aventail ExtraNet Servers, other SOCKS 5 servers, SOCKS 4 servers, or HTTP proxies.

Using an HTTP proxy server to control outbound traffic eliminates the need to install a separate SOCKS server. This HTTP proxy can filter outbound connection requests and route those requests to the specified servers. MultiProxy supports RFC 2068 HTTP Basic (username/password) authentication. If your proxy uses HTTP Basic (username/password) authentication, Aventail Connect will store the username and password information in the credential cache, as it does with SOCKS servers.

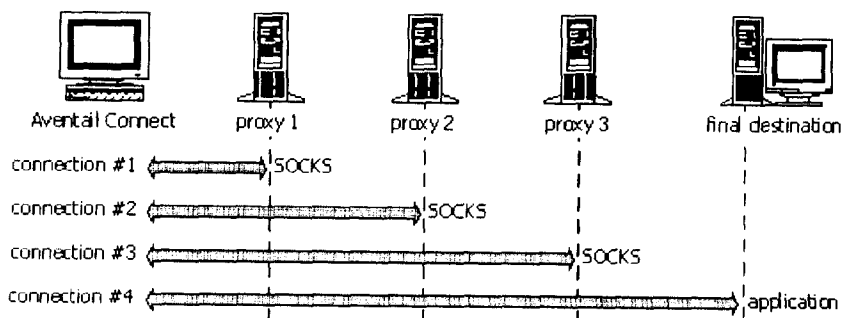


**NOTE:** The MultiProxy feature supports the use of HTTP proxies in Aventail Connect 3.1 only. HTTP proxies cannot be used in Aventail Connect 2.6.

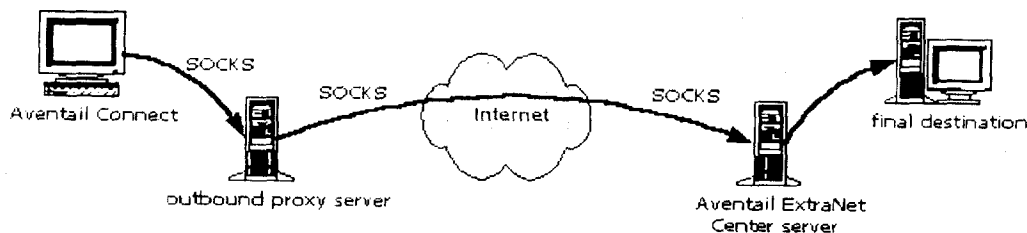
The steps for making a connection using MultiProxy are:

1. The client application requests access to the destination server.
2. Aventail Connect establishes a connection with the outbound server (SOCKS server or HTTP proxy). Aventail Connect then sends the access request to the outbound server, specifying the Aventail ExtraNet Server as the destination. The user authenticates with the outbound server, if necessary.
3. Aventail Connect instructs the outbound server to establish a connection with the Aventail ExtraNet Server on the specified port. The user authenticates with the Aventail ExtraNet Server, if necessary.
4. Aventail Connect instructs the Aventail ExtraNet Server to proxy its connection to the final destination.
5. Once the connection between the client and the Aventail ExtraNet Server is established, the outbound server simply relays the data.

The following example illustrates the connections made during a MultiProxy connection through three proxy servers.



In the following diagram, the Aventail ExtraNet Server acts as both a *destination* and a *server*. It is a destination because a proxy server routes traffic to it. It is a server because it routes traffic to the final destination.





**CAUTION:** *If using an HTTP proxy, you must configure your HTTP proxy and firewall to allow HTTPS/SSL connections to port 1080, OR you must run the Aventail ExtraNet Server on port 443 or port 563.*

### **Configuring Aventail MultiProxy**

You have two options for configuring MultiProxy. You can configure Aventail Connect 3.1 to redirect all Internet traffic (including extranet traffic) through your outbound proxy, or you can configure Aventail Connect 3.1 to redirect only extranet traffic through your outbound proxy.

#### **To configure Aventail MultiProxy**

1. Create a destination ("Final destination").
2. Create a server ("Extranet server").
3. **To redirect only extranet traffic:** Create a destination ("Extranet server"), using the same information from step 2, above.

-OR-

**To redirect all Internet traffic (including extranet traffic):** Create a destination ("Local network," the network local to Aventail Connect).

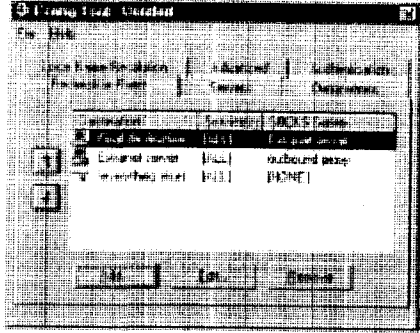
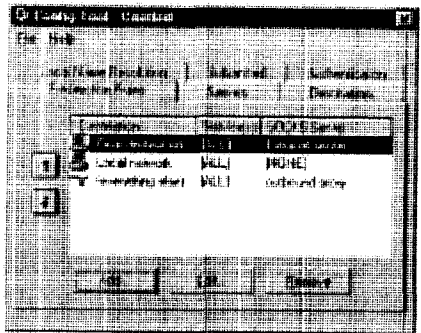


**NOTE:** *If you have multiple domains or subnets, you may need to create multiple destinations.*

4. Create a server ("Outbound proxy"). This can be a SOCKS 5, SOCKS 4, or HTTP proxy server.
5. Create a redirection rule (Redirect "Final destination" through "Extranet server").
6. **To redirect only extranet traffic:** Create a redirection rule (Redirect "Extranet server" through "Outbound proxy"). Do not redirect "(everything else)."

-OR-

**To redirect all Internet traffic (including extranet traffic):** Create a redirection rule (Do not redirect "Local network"). Redirect "(everything else)" through the outbound proxy. (NOTE: Your outbound proxy must belong to "Local network.")

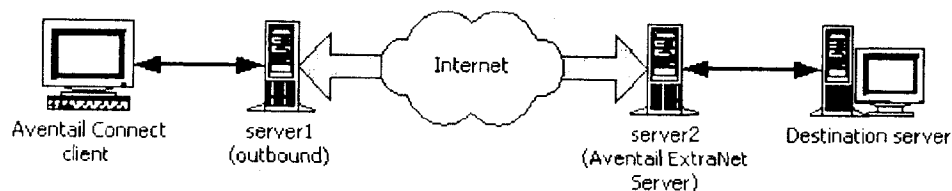
Redirect only extranet traffic	Redirect all Internet traffic (including extranet traffic)
	
<p>Redirect only the extranet traffic through the outbound proxy. Leave all other traffic alone.</p>	<p>Redirect all Internet traffic through the outbound proxy. Leave only "Local network" traffic alone.</p>

## PROXY CHAINING

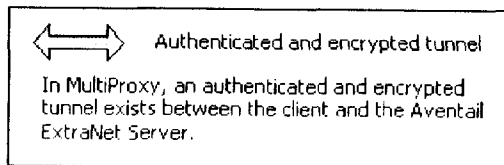
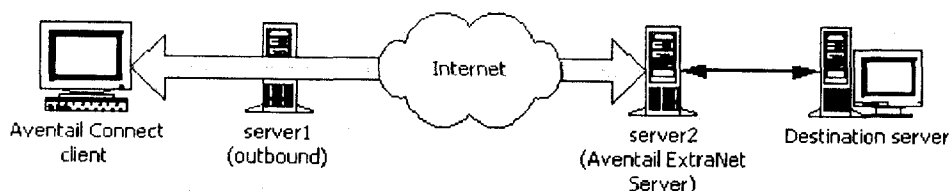
Proxy chaining is an Aventail ExtraNet Server feature. With proxy chaining, Aventail ExtraNet Servers forward connections for certain destinations to other proxy servers.

The following diagram and table illustrate the differences between MultiProxy and proxy chaining. In many cases, MultiProxy is the preferred method for traversing multiple firewalls. With MultiProxy, *each* proxy server can provide authentication, access control, and encryption.

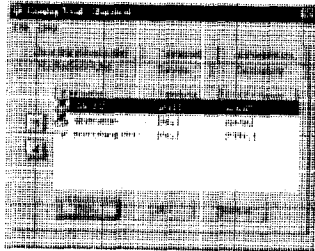
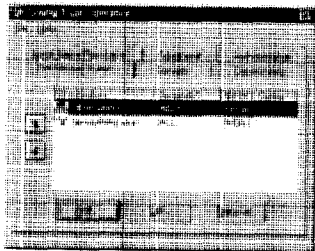
**PROXY CHAINING:** Server1 appears as a user to server2.



**MULTIPROXY:** The user authenticates with server2 directly.





Criteria	MultiProxy	Proxy Chaining
Server 1	Can be Aventail ExtraNet Server, other SOCKS 5 server, SOCKS 4 server, or HTTP proxy.	Must be Aventail ExtraNet Server.
Server 2	Must be Aventail ExtraNet Server.	Must be Aventail ExtraNet Server.
Authentication to Server 1	User authenticates (if necessary).	User authenticates.
Authentication to Server 2	User authenticates.	Server 1 authenticates automatically.
Trust model for Server 2	Not inherited. Each user must individually authenticate with Server 2.	Inherited from Server 1. Server 2 trusts everyone who authenticates to Server 1 equally.
Access control rules	Can be for specific users.	Treats everyone who authenticates to Server 1 equally.
Client configuration redirection rules		
Advantages	<ul style="list-style-type: none"> <li>• Server 1 can be an Aventail ExtraNet Server, other SOCKS 5 server, SOCKS 4 server, or HTTP proxy.</li> <li>• Most secure, because no security policy is inherited from Server 1.</li> </ul>	<ul style="list-style-type: none"> <li>• Client is aware of Server 1 only.</li> <li>• User authenticates only once, to Server 1.</li> </ul>
Disadvantages	<ul style="list-style-type: none"> <li>• User may need to authenticate more than once.</li> <li>• Client must be aware of Server 1 and Server 2.</li> </ul>	<ul style="list-style-type: none"> <li>• All users connecting through Server 1 appear as a single user to Server 2.</li> </ul>

## HTTP PROXIES AND WEB BROWSERS

Extranets often include Web pages that must be viewed with a Web browser. When a Web browser uses an HTTP proxy server, Aventail Connect sees connections being made to the HTTP proxy rather than to the final destination. Therefore, Aventail Connect cannot redirect the connections to the Aventail ExtraNet Server or provide authentication and encryption. For Aventail Connect to function properly, the Web browser cannot use the HTTP proxy to connect with sites protected in the extranet; this is because Aventail Connect must redirect and encrypt connections. The Web browser can still use the HTTP proxy to connect to sites that are not protected in the extranet.

If access to Web pages behind the Aventail ExtraNet Server requires users to connect through a Web browser (e.g., Microsoft Internet Explorer or Netscape Navigator), you must configure the Web browser to not use the HTTP proxy in the Web browser for those sites protected in the extranet.

When users need to access Web pages behind an Aventail ExtraNet Server, you must properly configure the Web browser.

### **Configuring Aventail Connect and the Web Browser**

There are two approaches to configuring Aventail Connect for use with a Web browser.

- Configure the Web browser to not use the HTTP proxy for any traffic. (Aventail Connect redirects all connections through the outbound proxy.)

-OR-

- Configure the Web browser to not use the HTTP proxy for only those sites that are protected in the secure extranet. (Aventail Connect redirects only extranet connections through the outbound proxy.)

To use either approach, you must first configure Aventail Connect. The Aventail Connect configuration is the same for both approaches, whether you are configuring your browser to not use the HTTP proxy for all traffic or for protected sites only.

#### **To configure Aventail Connect for use with a Web browser**

1. In the **Servers** tab of the Config Tool, add the HTTP proxy as a server.
2. In the **Destinations** tab of the Config Tool, add the HTTP proxy as a destination.
3. In the **Redirection Rules** tab of the Config Tool, edit the "(everything else)" rule to redirect all traffic to the HTTP proxy server.
4. In the **Redirection Rules** tab, select the HTTP proxy and select the **Do not redirect** option.



**CAUTION:** *Make sure you do not redirect the outbound proxy. Redirecting the outbound server or proxy will instruct the outbound proxy to redirect traffic to itself, causing Aventail Connect to behave unpredictably.*

#### **To configure the Web browser to not use the HTTP proxy for all traffic**

After you have configured Aventail Connect by following the instructions above, configure the Web browser by using one of the following procedures.

- **Microsoft Internet Explorer**
  - a. On the **View** menu, click **Internet Options**.
  - b. Click the **Connection** tab.
  - c. Click to clear the **Access the Internet using a proxy server** check box.
- **Netscape Navigator**
  - a. On the **Edit** menu, click **Preferences**.
  - b. Under "Category," click to expand **Advanced**, and then click **Proxies**.
  - c. Select **Direct Connection to the Internet**, and then click **OK**.

#### **To configure the Web browser to not use the HTTP proxy for protected sites only**

After you have configured Aventail Connect, configure the Web browser by using one of the following procedures.

- **Microsoft Internet Explorer**
  - a. On the **View** menu, click **Internet Options**.
  - b. Click the **Connection** tab.
  - c. Under "Proxy Server," click **Advanced**.
  - d. In the **Exceptions** box, type the URL of each site that is in the protected extranet.
- **Netscape Navigator**
  - a. On the **Edit** menu, click **Preferences**.
  - b. Under "Category," click to expand **Advanced**, and then click **Proxies**.
  - c. Select **Manual Proxy Configuration**, and then click **View**.
  - d. In the **Exceptions** box, type the URL of each site that is in the protected extranet.

## CONFIGURING THE HTTP PROXY

To allow SSL connections to destination ports other than 443 (https) and 563 (snews), you may need to configure your HTTP proxy. Typically, if you plan to connect to a SOCKS server on port 1080 using an HTTP proxy, you must change the HTTP proxy configuration.

To avoid changing the HTTP proxy configuration, you must run the destination Aventail ExtraNet Server on port 443 or port 563, and configure Aventail Connect accordingly.

Most HTTP proxies can allow connections to port 1080. The following instructions describe how to configure the Microsoft Proxy Server, Netscape Proxy Server, or Apache Web Server to allow port 1080 connections.

- **Microsoft Proxy Server 2.0:** Follow the Microsoft instructions at <http://support.microsoft.com/support/kb/articles/q1840/0/28.asp>. You must modify a registry setting with `regedt32.exe`. (`regedit.exe` will not work; you must use `regedt32.exe`.)
- **Netscape Proxy Server 3.5:** Add the following to your `obj.conf` file:  

```
<Object ppath="connect://*"> (all ports)
Service fn="connect" method="CONNECT"
</Object>
```

 To specify a particular port, add the following to your `obj.conf` file:  

```
<Object ppath="connect://*:1080"
```
- **Apache Web Server 1.3.2 (Linux) with Proxy Support:** The following two lines must be included in the `httpd.conf` file:  

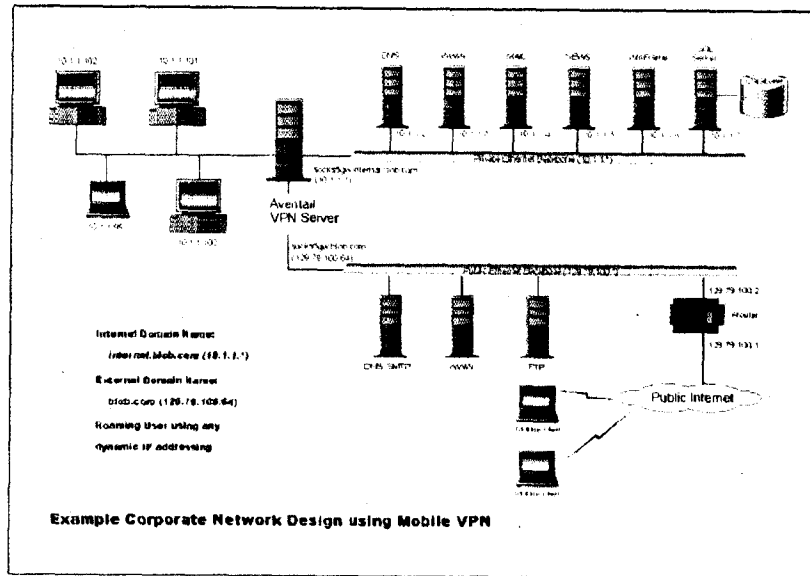
```
ProxyRequests On
AllowCONNECT <port list> (NOTE: This feature is available only
on version 1.3.2 and greater.)
```

## EXAMPLE NETWORK CONFIGURATION

The following section describes the setup of Aventail Connect in an example network configuration using the Aventail ExtraNet Server.

### CONFIGURATION USING AVENTAIL EXTRANET SERVER

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



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 ExtraNet 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 ExtraNet 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 Aventail ExtraNet Server.

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.

The Aventail ExtraNet 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.



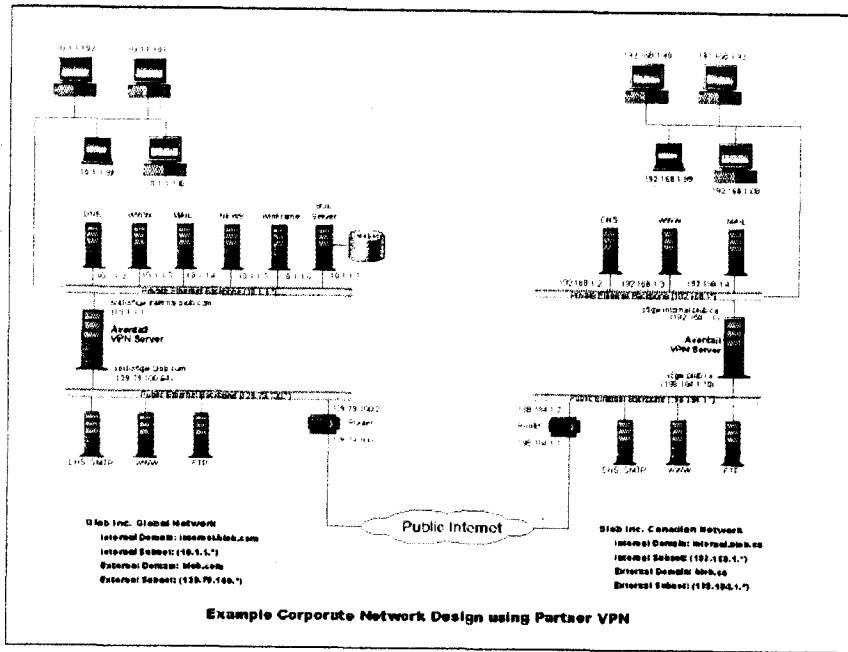
**CAUTION:** *Since the internal network address space is part of the IANA reserved address space (per BCP RFC 1918) routing **MUST** be disabled on this host and routing advertisements for this internal network **MUST NOT** be propagated to the outside world.*

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.



**SEE ALSO:** *For additional information on how to configure the Aventail ExtraNet Server product, consult the Aventail ExtraNet Server Administrator's Guide.*

Installing and using Aventail Connect for remote access purposes differs a bit from its installation and use within a corporate network. First, configuration files need to be kept locally on the workstation or laptop. This is due to the inability to share a file server that allows direct access outside the perimeter of the private network. Second, not all traffic passes through to the Aventail ExtraNet Server. Only traffic destined for the internal network is authenticated and encrypted; all other traffic passes through Aventail Connect unchanged. For instance, browsing the Internet from the mobile user workstation occurs as if Aventail Connect is not even running in the background. Large sites with many mobile users will want to set up an internal file server for a network installation for all mobile users to easily install and configure Aventail Connect. For more information, refer to "Network Installation."



## Utilities Reference Guide

This section explains:

- Commands on the System menu, including Close, Hide Icon (in Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51), Help, About, Credentials, and Configuration File
- How to use the Aventail Connect utilities, including the Config Tool, the Logging Tool, and S5 Ping, all displayed through the Utility Programs menu.
- How to use Secure Extranet Explorer (SEE)/Extranet Neighborhood.

### SYSTEM MENU COMMANDS

Even though Aventail Connect requires little to no interaction with the user, there are commands on the Aventail Connect System menu. To display the System menu, right-click the **Aventail Connect** icon in the system tray on the taskbar (Windows 95, Windows 98, and Windows NT 4.0) or click the minimized **Aventail Connect** icon (Windows 3.1, Windows for Workgroups 3.11, or Windows NT 3.51).

#### Aventail Connect System Menu Commands

Menu Command	Function
Close	Closes Aventail Connect.
Hide Icon	Hides the <b>Aventail Connect</b> icon from view. Not available in Windows 95, Windows 98, and Windows NT 4.0.
Help	Accesses Help.
About	Displays Aventail Connect <b>About</b> box.
Credentials	Displays authentication credentials.
Configuration File	Selects new configuration file via <b>Aventail Connect Configuration File</b> dialog box.

Each of the commands is discussed below.

#### CLOSE

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



## HIDE ICON

This command hides the **Aventail Connect** icon from view (Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51 only). Aventail Connect will run in the background. *The **Hide Icon** command is not available in Windows 95, Windows 98, and Windows NT 4.0.*

## HELP

This command accesses Aventail Connect Help.

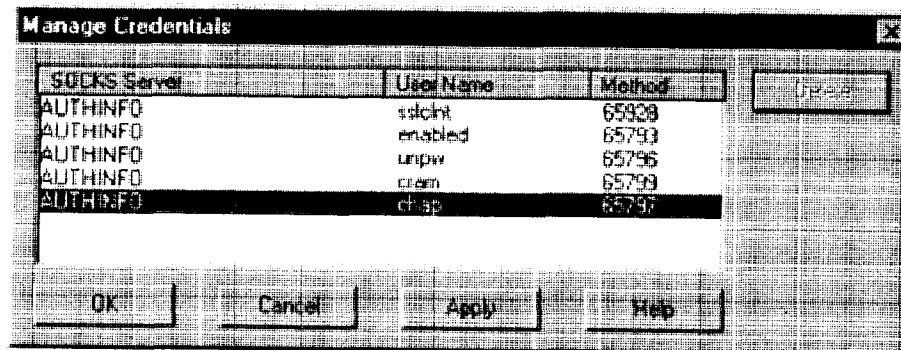
## ABOUT

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

## 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 an extranet (SOCKS) server requiring user authentication. (Aventail Connect prompts you with an authentication dialog box.) As long as your credentials are in memory, you can establish connections to associated extranet servers without needing to reenter your authentication information.

You cannot edit credential data fields; you can, however, delete individual credential entries. Aventail Connect will prompt you to enter updated authentication information when you reestablish a connection to the associated extranet server.





**NOTE:** You cannot edit the "AUTHINFO" entries in the **Manage Credentials** dialog box. This information is for diagnostic purposes only.

Field	Definition
SOCKS Server	Extranet (SOCKS) server name.
User Name	User name for the extranet server.
Method	Authentication method.

#### To delete a credential entry

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

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

This deletes the credential information.

#### To exit the Manage Credentials dialog box

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

-OR-

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

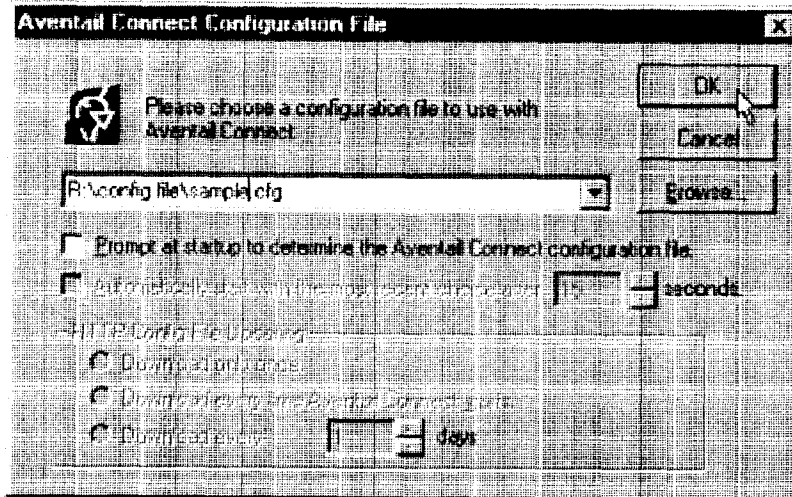


**NOTE:** Clicking **Apply** saves changes but keeps the dialog box open so you can keep working.

## CONFIGURATION FILE

This command lets you load a different configuration file via the **Aventail Connect Configuration File** dialog box. Aventail Connect 3.1 allows you to use a new or modified configuration file immediately, without needing to restart Aventail Connect and any Aventail-processed applications.

For more information about the configuration file, refer to "Configuring Aventail Connect."



**To load a configuration file**

- Select the configuration file you want to load (use the **Browse** button), and then click **OK**.
- If you want Aventail Connect to start automatically with your most recent choice of configuration file, select the **Automatically start...** check box, and then select the start delay (in seconds).

The new configuration file transparently loads into Aventail Connect. You can close and restart Aventail Connect for your change to take effect, or wait the specified length of time if you selected the **Automatically start...** checkbox.

**UTILITIES**

To display the Utility Programs menu, right-click the **Aventail Connect** icon in the system tray on the taskbar (Windows 95, Windows 98, or Windows NT 4.0) or click the minimized **Aventail Connect** icon (Windows 3.1, Windows for Workgroups 3.11, or Windows NT 3.51).

**Aventail Connect Utility Program Menu Commands.**

Menu Command	Function
Config Tool	Runs the Config Tool. (Optional)
Logging Tool	Runs the Logging Tool. (Optional)
S5 Ping	Runs the ping and traceroute utilities. (Optional)

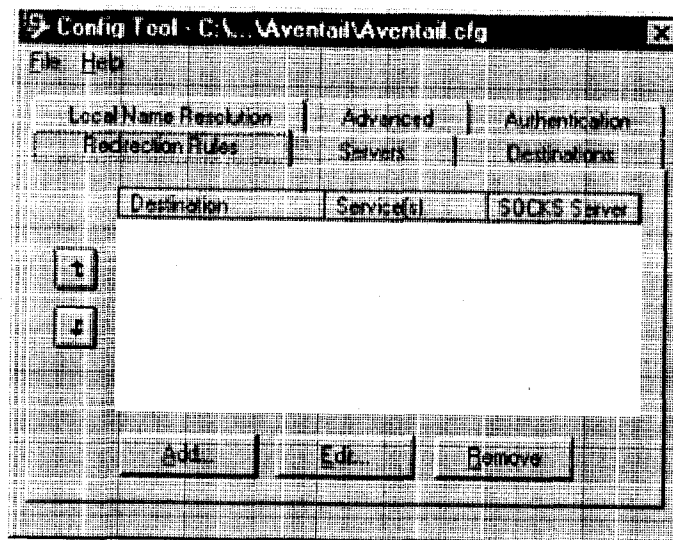
Each of the commands is discussed below.



**NOTE:** The *Config Tool*, *Logging Tool*, and *S5 Ping* commands are optional components and will only appear when the network administrator has included them in a custom setup package. They are discussed in the sections "Config Tool," "Logging Tool," and "S5 Ping."

## CONFIG TOOL

The Aventail Connect Config Tool creates configuration files that determine how network requests will be routed and which authentication protocols will be enabled. (This option may not be available to all users if the network administrator has chosen not to install it.)



Network administrators generally create configuration files during Aventail Connect installation. However, you can add, remove, or modify configuration files at any time. If necessary, you can create several configuration files 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. "Configuring Aventail Connect" discusses the Config Tool in detail.

## LOGGING TOOL

The Logging Tool is an optional diagnostic utility for tracing Aventail Connect and WinSock activity. When running a trace, the Logging Tool displays errors, warnings, and information as Aventail Connect generates them. You can save the message list to a log file that Aventail Technical Support can use in troubleshooting technical problems, including Aventail Connect network, extranet (SOCKS) server, and WinSock application interoperability problems. Aventail Technical Support engineers may request that you perform a verbose trace, log it to a file,

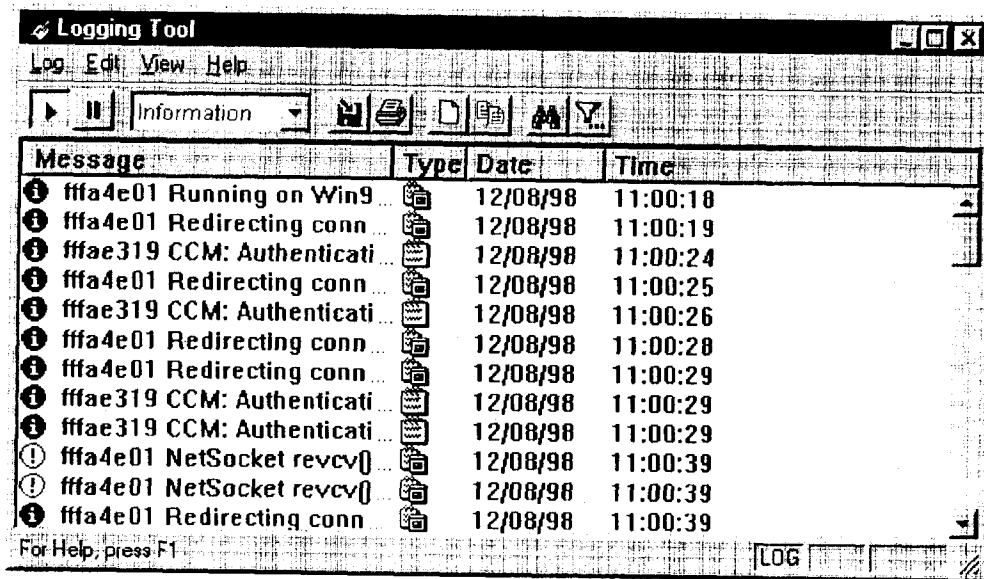
and e-mail it to them as an attachment. Log files are also useful when running Aventail Connect for the first time, to ensure that network traffic is being routed properly.

**To trace Aventail Connect activity**

1. Windows 95, Windows 98, or Windows NT 4.0: Either right-click the **Aventail Connect** icon (in the system tray on the taskbar) and click **Logging Tool**, or select **Start | Programs | Aventail Connect | Logging Tool**.

-OR-

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



2. In the **Log** menu, click **Level** and select one of the five levels of information you want to trace.

-OR-

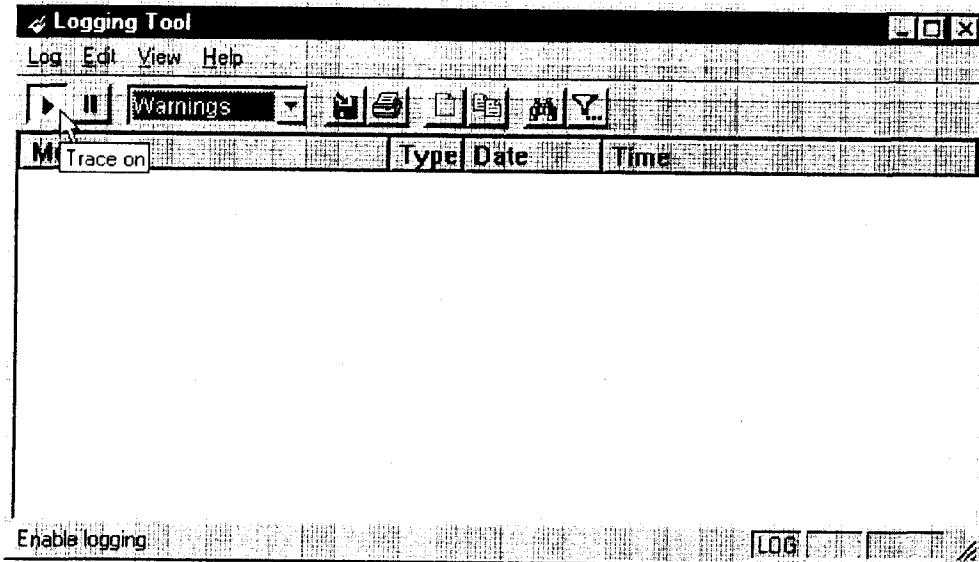
Select one of the five levels from the drop-down list on the toolbar.

Select	To Log
Fatal Errors	Fatal errors only
Errors	Errors and fatal errors only
Warnings	Errors and warnings only
Information	Errors, warning, and information
Verbose	All of the above, and more descriptive information on progress of connections

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

-OR-

Click the **Trace On** button on the toolbar (shown below).

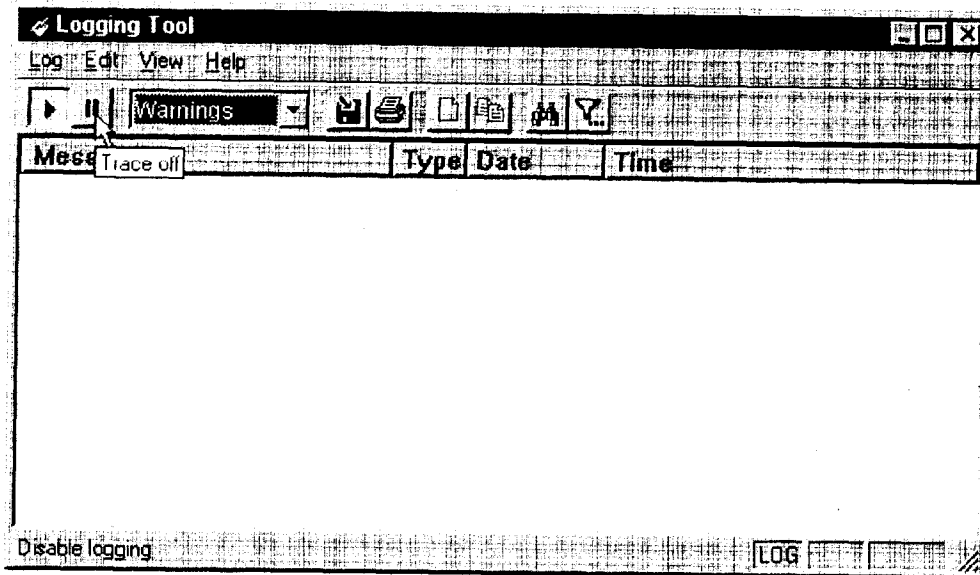


The log window will now record and display trace information as it is generated by Aventail Connect. You can tell when the trace function is active because messages are scrolling down the screen and the **Trace On** button is depressed.

4. When you are ready to stop the Trace function, click **Trace** on the **Log** menu.

-OR-

Click the **Trace Off** button on the toolbar (shown below).



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

#### To save a log file

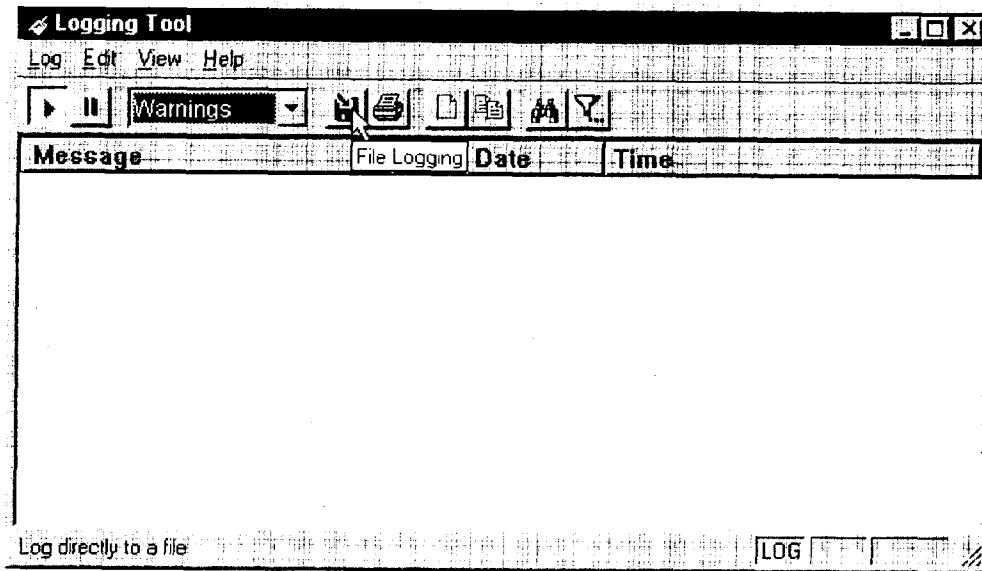
The Logging Tool allows you to append each new message to the end of a .LOG file during the trace, or save the contents of the log window at any time. If you save during a trace, Aventail Connect will append messages to the log file until you stop the log function. You must save data in the log window to retain it.

You cannot open a preexisting log file from within the log window. To open a preexisting log file, you must open it in a text editor such as Notepad.

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

-OR-

Click the **File Logging** button on the toolbar (shown below).



2. Enter the filename in the **Select Log File** dialog box.
  - To save the contents of the log window at any time, click **Save As** on the **Log** menu and then enter the filename.

#### To filter messages in the log window

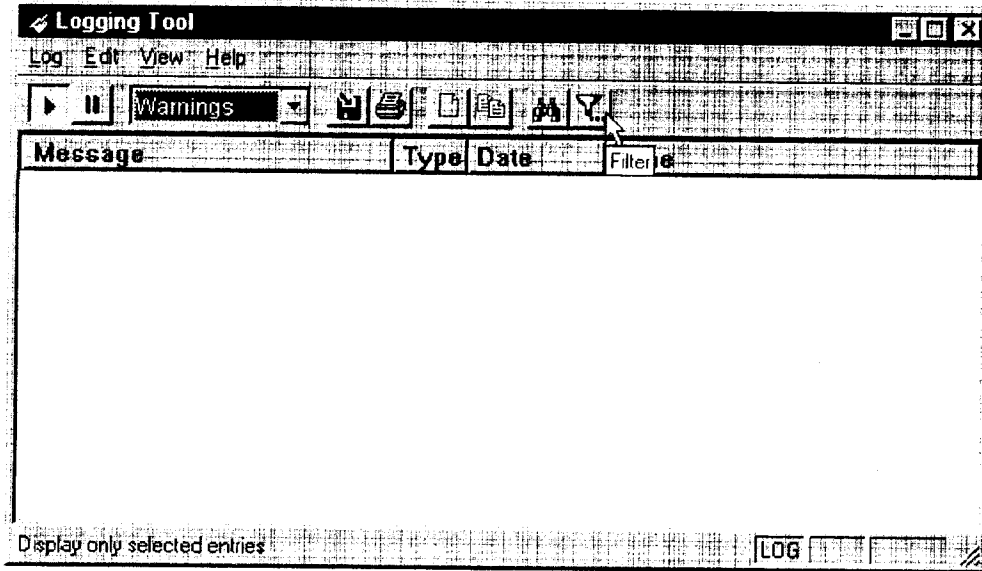
You can filter the contents of a log window by selecting the types of messages you want to view. By selecting a specific type of message, you can easily scan the information on-screen. If you save data to a log file, a view filter will not affect the file contents; it merely adjusts the screen display of those contents.

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

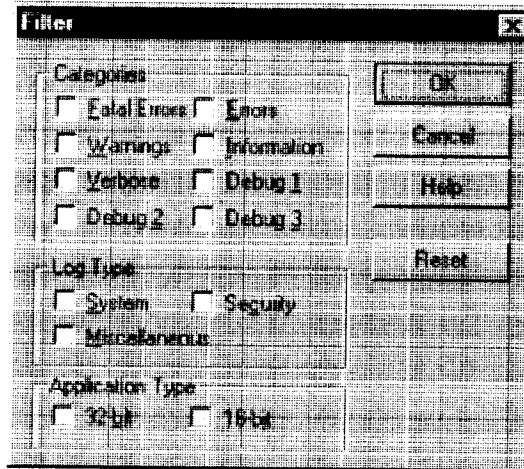
-OR-

Click the **Filter** button on the toolbar (shown below) to display the **Filter** dialog box.





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





Field	Definition	
Categories	Select any of the five filters to display errors, fatal errors, warnings, information and/or verbose information in the log window.	
Log Type	Select the type of log to be filtered. (Currently, the only valid log type used in Aventail Connect is Miscellaneous.)	
Application Type*	32-bit	Show messages from 32-bit applications.
	16-bit	Show messages from 16-bit applications.
	*These options are disabled if you are running 16-bit Windows.	

2. Under "Categories," select one or more of the five filter check boxes. The log window will adjust the display based on your selection(s).
3. Under "Log Type," select the log type to filter.
4. Under "Application Type," select one or both of the check boxes.

**To change the view parameters**

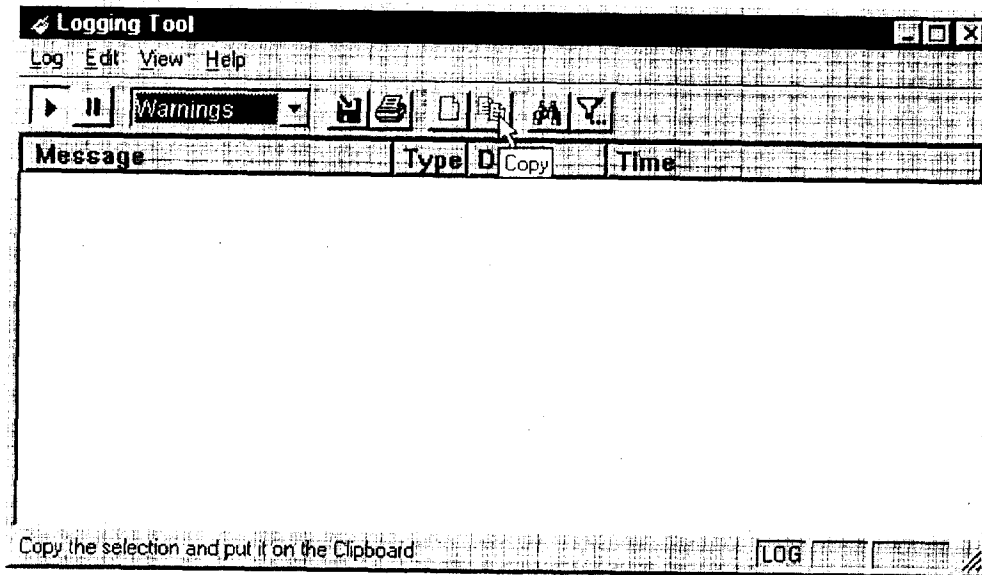
The display font and window options can be customized as follows:

- On the **View** menu, click **Font**. Enter your font preferences into the standard **Windows Font** dialog box.
- To display or hide the toolbar and status bar, click **Toolbar** and/or **Status Bar** on the **View** menu.

**To copy the log window**

You can copy the log window contents to the Windows Clipboard.

- To copy all of the log window contents to the Windows Clipboard, click **Select All** on the **Edit** menu. Then click **Copy** on 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 select **Copy** on the **Edit** menu or click the **Copy** button on the toolbar.



#### To print the log window

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

- On the **Log** menu, click **Print**.

-OR-

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

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

#### To find a specific message

The **Find** command 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.

- On the **Edit** menu, click **Find**.

-OR-

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

Then enter your search parameters in the **Find** dialog box.

#### To clear the log window

Clear the log window contents when you are ready to execute a new trace.

- On the **Edit** menu, click **Clear All**.

-OR-

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

#### To close the log window

When you are ready to close the log window, make sure you have saved the contents of the trace for later reference. All settings are saved when you exit.

- On the **File** menu, click **Exit**.

## S5 PING

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

- 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, SOCKS v5 does not support ICMP. Because ping and traceroute are based on ICMP, there is no way to directly proxy a ping or traceroute request. To circumvent this problem, Aventail Connect provides a utility called S5 Ping.

S5 Ping determines whether a host outside of an extranet server is active. After a response from the host returns, the extranet server relays the data back to the client and displays it in the **S5 Ping** dialog box.

#### To launch S5 Ping

You can use S5 Ping whether or not Aventail Connect is running. However, if the server that you are connecting through requires authentication, you must load Aventail Connect before reconnecting.

- Windows 95, Windows 98, or Windows NT 4.0: Select **Start | Programs | Aventail Connect | S5 Ping**.

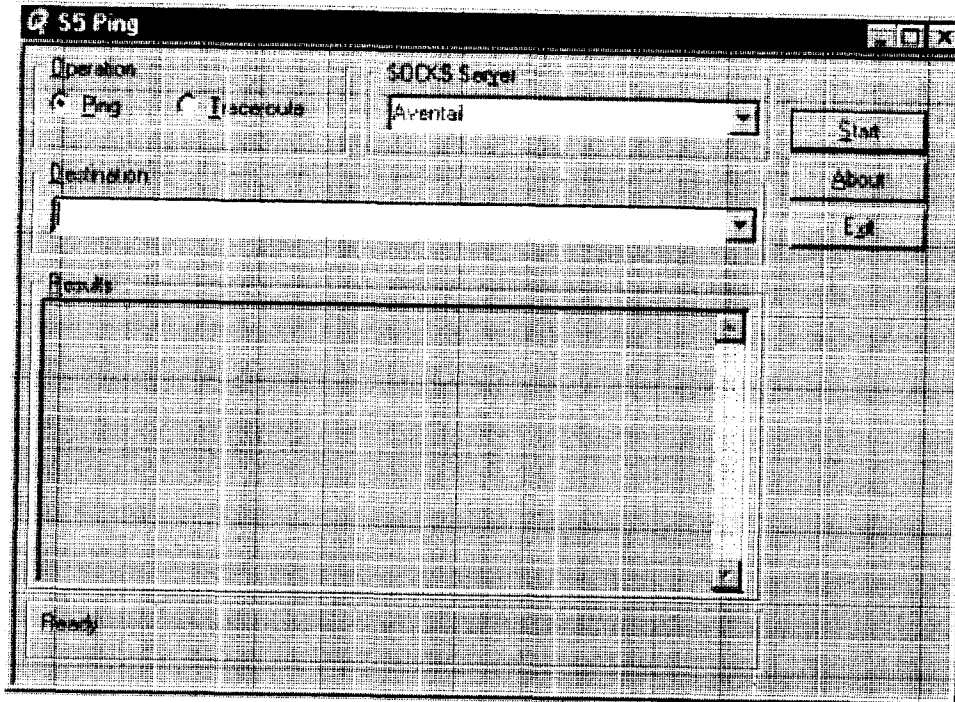
-OR-

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

-OR-

If Aventail Connect is already running, right-click the **Aventail Connect** icon on the taskbar and click **S5 Ping** (Windows 95, Windows 98, or Windows NT 4.0), click the minimized **Aventail Connect** icon in the System menu (Windows 3.1, Windows for Workgroups 3.11, or Windows NT 3.51).

The **S5 Ping** dialog box appears.



**NOTE:** S5 Ping will function without a properly configured Aventail Connect; however, the user will be required to type the information about the target extranet server and target host into the **SOCKS Server** and **Destination** boxes.

Field	Definition
Operation	Select ping or traceroute.
SOCKS Server	The Extranet (SOCKS) server that will execute the operation. If Aventail Connect is already configured, this list will be preloaded with extranet servers from the configuration file.
Destination	The extranet server you want to ping (or traceroute). If Aventail Connect is already configured, this list will be preloaded with single host destinations defined in the configuration file. (See "Configuring Aventail Connect.")
Results	The results of successful connection. The format of the results will vary based upon the extranet server platform.

S5 Ping can be used whether or not Aventail Connect is running. However, if the server that you are connecting through requires authentication, you must load

Aventail Connect before connecting. The network administrator may or may not make S5 Ping available to users during installation. In some cases, the **S5 Ping** command will not appear on the Aventail Connect System menu or in the program group.

Once the **S5 Ping** dialog box 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 an Aventail ExtraNet Server to carry out the operation. If no servers are listed (because S5 Ping did not locate an Aventail Connect configuration file), type the extranet 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 Aventail Connect configuration file), type the hostname or IP address of the host you want to ping or traceroute.
4. Click **Start** to execute the operation. **Start** then changes to **Stop**. Results from any previous operation are cleared from the window.
5. If the extranet 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 "Manage Authentication Modules" in the *Administrator'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 **Stop**.

This stops the operation and changes **Stop** to **Start**. The results of the operation remain displayed in the **S5 Ping** dialog box.

#### To exit S5 Ping

---

- Click **Exit**.

This clears the results and closes the **S5 Ping** dialog box.

## 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. 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.



**NOTE:** Some installations of Aventail Connect may not include SEE. Network administrators can decide whether or not to include SEE in a custom setup package.

Extranet Neighborhood, a Windows Explorer shell extension, is a collection of Windows file servers and Windows NT domains. Network Neighborhood displays only those remote computers that the network administrator has specified. SEE requires a hosts file (SEEHosts) that determines which Windows file servers and NT domains are available. You can include a SEEHosts file with the Aventail Customizer tool. If users install a custom package that does not include a SEEHosts file, then the first time they open Extranet Neighborhood, SEE will create a SEEHosts file. For more information, see the "Customizer" section in the *Administrator's Guide*.

Extranet Neighborhood offers Aventail Connect users a secure alternative to traditional file-browsing methods. Users can securely access computers from the desktop through Extranet Neighborhood (see icon below), or through Windows Explorer.



Generally, you will use Extranet Neighborhood to connect to a remote network through Aventail Connect. For example, you will use Extranet Neighborhood when:

- you are inside the office, on the corporate network, and you connect through an Aventail ExtraNet Server to your company's remote site, or to another company's network.
- you are outside the office, and you connect your laptop through an Aventail ExtraNet Server to your internal company network, or to another company's network.



**NOTE:** To use Extranet Neighborhood with remote hosts, Aventail Connect must be running and configured correctly.

## HOW EXTRANET NEIGHBORHOOD WORKS

Typically, with Windows networking, the Microsoft Windows Explorer and Network Neighborhood browse files using NetBIOS (NBT), over TCP. Network Neighborhood does not use the standard WinSock programming interface. This prevents Aventail Connect from redirecting TCP connections. Since Aventail Connect redirects only WinSock calls, it cannot redirect NBT calls.

To deliver a secured version of standard Windows browsing, Aventail Connect redirects NBT calls to WinSock. This allows Aventail Connect to redirect this traffic based on a set of redirection rules, as defined in the Aventail Connect configuration file.

Extranet Neighborhood can use either hosts files or Windows Internet Naming Service (WINS) servers to map a computer's Internet (host) name to its Windows machine name. Without a hosts file or a WINS server, Extranet Neighborhood cannot associate a computer's Internet name with its Windows machine name.

Extranet Neighborhood includes a browsing mode, which allows you to view a dynamic list of available Windows hosts. Hosts files provide a static list of hosts.

There are two basic methods for configuring Extranet Neighborhood.

- **Listing WINS Servers:** List only WINS servers for the domain(s) in the hosts file. You do not need to list individual hosts within the domain.
- **Listing Individual Hosts:** List every individual host in the hosts file that will be accessible to users.

### LISTING WINS SERVERS

To use Extranet Neighborhood in the browsing mode, you must configure Extranet Neighborhood to use WINS, and you must identify the IP address (host-name) of the WINS server(s) and, possibly, the primary domain controller (PDC) for the domain. If you do not specify a WINS server, you will not be able to use Extranet Neighborhood in the browsing mode.

The PDC for the domain is required only if the destination network is not accessible by UDP. (For example, when using MultiProxy, the destination network is not UDP-accessible.) When Extranet Neighborhood is in browsing mode, it must be able to resolve the name of the host. If the destination network is UDP-accessible, then the WINS server is used to map a computer's Internet (host) name to its Windows machine name. If the destination network is not UDP-accessible, then Extranet Neighborhood uses the PDC and DNS to determine the host's address.



## LISTING INDIVIDUAL HOSTS

To use Extranet Neighborhood in the static host list mode, you must define, in the hosts file, each individual host in the domain. This allows you to restrict access to designated hosts only. In the hosts file, you must specify the host's IP address or DNS name along with the Windows machine name. WINS and PDC are not used in this method.

## INSTALLING EXTRANET NEIGHBORHOOD

When installed, Extranet Neighborhood appears on your desktop as an icon, and in Windows Explorer. You can open, move, copy, and delete files in Extranet Neighborhood just as you would in Network Neighborhood.

If you need to install Extranet Neighborhood, install it from the Aventail Connect CD. Or, if you downloaded your copy of Aventail Connect, run the downloaded executable package. When the **Installation Components and Sub-components** dialog box appears, select **Extranet Neighborhood** (located under **Components**). Continue with the installation process.

The default installation directory is  
 \Program Files\Aventail\Connect.



**NOTE:** *Secure Extranet Explorer/Extranet Neighborhood is available only on Windows 95, Windows 98, and Windows NT 4.0 operating systems.*

## CONFIGURING EXTRANET NEIGHBORHOOD

You can include a SEEHosts file with the Aventail Customizer tool. Only by installing a custom package will users have a local or remote hosts file automatically configured. If users install a custom package that does not include a SEEHosts file, the SEE Configuration wizard will run when users open Extranet Neighborhood for the first time. The SEE Configuration wizard walks you through the process of defining local or remote hosts files. Aventail recommends that you use the Customizer tool to distribute Extranet Neighborhood, bundled with a hosts file, in a custom setup package.

Extranet Neighborhood can automatically construct a hosts file from your local network or a remote network. Using the Search feature, Extranet Neighborhood can automatically "browse" available computers and build the local hosts file. The Search feature is available through the **Extranet Neighborhood Properties | Local** tab. Alternatively, you can enter the names of the available computers manually. The Search feature browses only those computers that are within your internal network. To search remote networks, you must manually enter the fully qualified hostname of each remote WINS server that is outside your Aventail ExtraNet Server. When using the Search feature, the same UDP restrictions described in "Listing WINS Servers" apply.



**NOTE:** To use the Search feature, Aventail Connect must be running and configured correctly.

Do not use the Search feature if you are using the WNS-browsing mode. The Search feature builds the local hosts file for all of the computers, which is not necessary with WNS. Use Search when creating a local hosts file using the "listing individual hosts" method.



**NOTE:** When you click **Search**, you may see more than one domain in the resulting local hosts file. This is because Search includes trusted domains.

### To create a hosts file

Use this procedure if you have not yet created a hosts file.

1. Decide which method, listing WNS servers or listing all individual hosts, to use.
2. If no hosts file exists, launch Extranet Neighborhood (Extranet Neighborhood will prompt you automatically if you are running Extranet Neighborhood for the first time),

-OR-

Right-click the **Extranet Neighborhood** icon on your desktop and then click **Properties**.

3. Follow the on-screen instructions to create the hosts file.
4. To distribute the new hosts file, include the SEEHosts file in your custom setup package, if using the Customizer tool.

After creating the hosts file, users can browse only those domains and machines that the network administrator has included in that list of hosts. This list may be a local hosts file called "SEEHosts" and/or a remote host list, which is identified by [share]\[path]\[filename].



**NOTE:** To use the browsing mode, you must specify the domain's WINS server(s) in the local hosts file.



**CAUTION:** SEE cannot recognize share names that contain special characters (e.g., é) or multiple spaces (e.g., Aventail Custom Computer). SEE also will not recognize hidden one-letter share names (e.g., C\$ or D\$).

## SEE CONFIGURATION METHODS

There are numerous methods for configuring SEE. The three most common methods are described below.

### Local Hosts File Method

With this method, the hosts file contains a list of all domains and servers in the local hosts file. Every host is listed.

There are two ways to configure SEE using this method.

- In the **Extranet Neighborhood Properties | Local** tab, manually add each domain and host to the local hosts file
- OR-
- On the **Local** tab, click **Search**, click **Search Local Network**, and then search any remote networks, if necessary. SEE automatically builds a list of all hosts. You may delete hosts from the local hosts file if you do not want users to view them.



**NOTE:** To view your domains, double-click the **Extranet Neighborhood** icon on your desktop. If you make changes to the hosts file, you can reload the **Extranet Neighborhood domains** window by pressing the **F5** key.

### Remote Hosts File Method

With this method, the local hosts file contains the path of the remote hosts file, and the remote hosts file contents are determined by which configuration method you use.

To use this method, first create the remote hosts file, and then create a local hosts file that points to the remote hosts file.

#### To configure SEE using the remote hosts file method

1. Create a local hosts file, using one of the methods listed above, and copy it to a central location. (This creates a remote hosts file; this file is not distributed with Aventail Connect.)
2. On the **Remote** tab, click **Add**, and then add a pointer to the remote hosts file that you created in Step 1. (This file is distributed with Aventail Connect.)



**NOTE:** You can point to multiple remote hosts files on a single list.

### WINS Browsing Method

With this method, the hosts file contains a list of all domains, and the WINS servers for each domain. You do not need to list all of the computers.

To use this method, add each domain in the **Local** tab, specifying the primary WINS server and, if applicable, the secondary WINS server, and then select the **Make domain browsable** check box in the **Windows Domain** dialog box.

### Choosing a Method

Each of the three methods has advantages and disadvantages. The table below lists pros and cons for each of the three methods.

Method	Advantages	Disadvantages
Local hosts file with individual computers	The administrator controls exactly which hosts the users can see. On slower connections, this method is fastest since you do not need to send a list of servers to the client.	The administrator must update the local hosts file if file servers are added to or removed from the domains.
Remote hosts file	<ul style="list-style-type: none"> <li>• The administrator can edit the centrally stored hosts file whenever necessary.</li> <li>• If the hosts file is stored behind a firewall, SEE can go through an extranet server (using encryption and authentication) to reach it.</li> </ul>	<ul style="list-style-type: none"> <li>• Users are immediately prompted to enter authentication credentials upon opening SEE (because SEE must load the remote hosts file).</li> <li>• If a user loses network connectivity to the hosts file, SEE will not display the list of hosts/computers.</li> </ul>
Local hosts file with WINS browsing	The administrator does not need to update the hosts file if new computers are added or removed.	<ul style="list-style-type: none"> <li>• The administrator must update the local hosts file if domains are added or removed.</li> <li>• The administrator cannot control which computers appear in SEE; all computers in the NT domain are displayed.</li> <li>• On slower connections, this method is slower than other methods because a list of computers must be sent to the client.</li> </ul>

You are not limited to using only one method for configuring SEE. You can use a combination of the various methods. For example:

- Use WINS browsing for some domains, and explicitly list hosts for other domains

-OR-

- Use multiple remote hosts files

-OR-

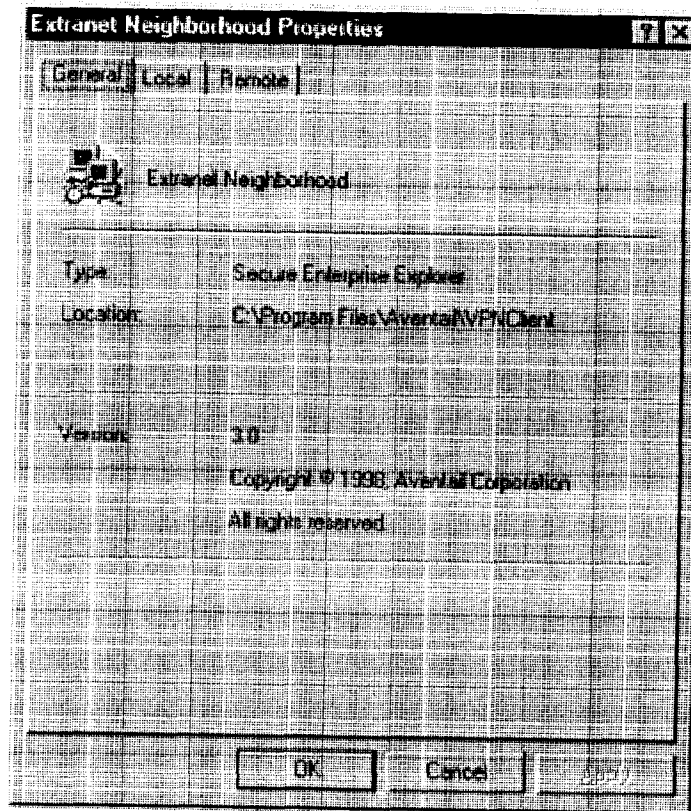
- Specify some computers in a local hosts file and others in a remote hosts file.

## SEE PROPERTIES

To access information about the current configuration of SEE, or to make changes to that configuration, right-click the **Extranet Neighborhood** icon and click **Properties**, or click **View | Options** in any open **SEE** window. The **Extranet Neighborhood Properties** window will appear with the **General** tab selected.

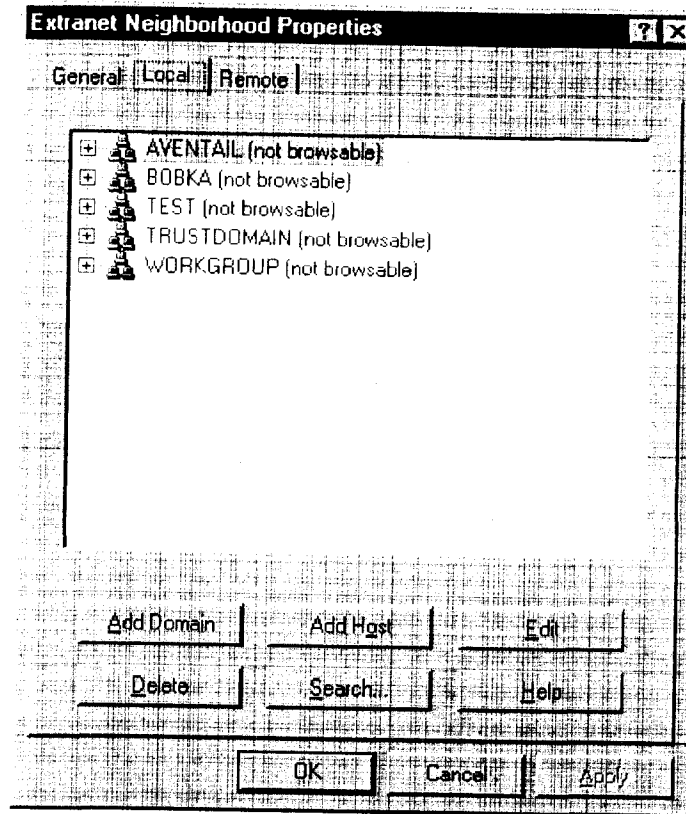
## THE GENERAL TAB

The **General** tab displays information about the current configuration of SEE.



## THE LOCAL TAB

The **Local** tab displays the computers that are listed in the local hosts file.



If you have specified a host in the local hosts file, you can add, edit, or remove computers or domains that appear in the **Local** tab. If you have specified hosts in the remote hosts file, they will not appear in this tab. To edit hosts in the remote hosts file, you must copy the file to your Aventail Connect directory, edit it, and then replace it in the remote hosts directory.

If you are using the WINS browsing mode, the individual computer names will not appear. Any hosts specified in remote hosts files, including WINS servers, will not appear in this tab.

The **Add Host** and **Add Domain** buttons allow you to add additional computers or domains in the **Add Host to Aventail** dialog box and the **Windows Domain** dialog box.

If no computers or domains appear in your **Local** tab, check the **Remote** tab. It is possible that your network administrator has configured Extranet Neighborhood with only a remote hosts file.

The **Search** feature can automatically browse available computers in local or remote domains and populate your local hosts file. Alternatively, you can enter the names of the hosts files manually.



**NOTE:** To view your domains, double-click the **Extranet Neighborhood** icon on your desktop. To reload the hosts files in the **Extranet Neighborhood domains** window, press the **F5** key.



**NOTE:** In the **Local** tab, "browsable" domains do not show individual computers in them.

### Hosts File Locking

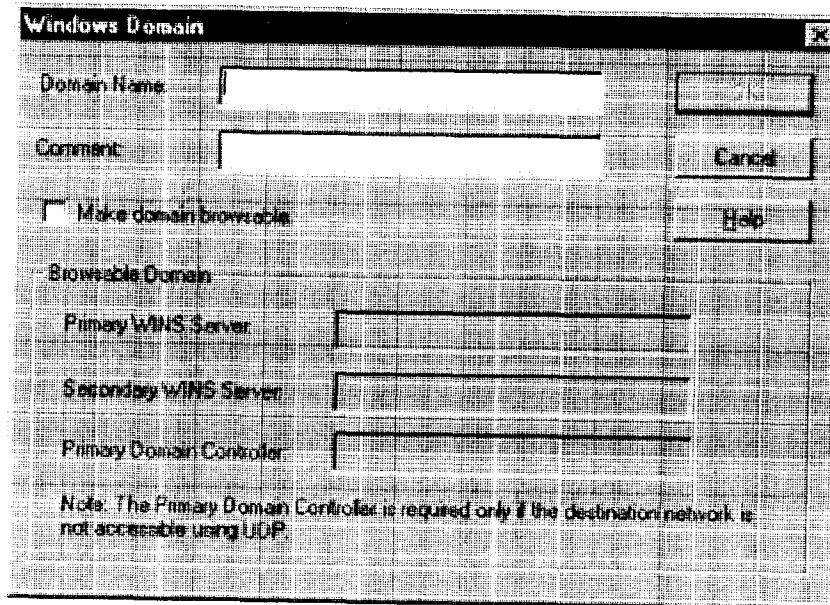
If the controls in this window are disabled (dimmed), then the hosts file has been "locked." The network administrator determines which, if any, hosts files are locked.

You can lock and unlock files from any **Extranet Neighborhood Properties** tab.

- To lock a file, use the **Ctrl+L** command.
- To unlock a file, use the **Ctrl+U** command.

### Windows Domain Dialog Box

To open the **Windows Domain** dialog box, click **Add Domain** in the **Extranet Neighborhood Properties | Local** tab.



For each domain, you can either specify the WINS server names or specify each individual host that should appear in the domain. Listing WINS servers will result in a smaller, more manageable hosts file. You must add a domain before you can add hosts to that domain.

To make the specified domain “browsable,” enter WINS server information in the **Primary WINS Server** box and, if desired, the **Secondary WINS Server** box. In both of these boxes, you can enter either the server’s IP address or its fully qualified host name. You must also select the **Make domain browsable** check box. If you do not select the **Make domain browsable** check box, Extranet Neighborhood will display only those computers in the local or remote hosts file, even if you have specified a WINS server.



**NOTE:** To use the browsing mode for a domain, you must specify the domain’s WINS server(s) in the hosts file. You must specify the WINS server(s) only if you want to use the browsing mode.

To view your domains, double-click the **Extranet Neighborhood** icon on your desktop. To reload the hosts files in this screen, press the F5 key.

### Add Host to Aventail Dialog Box

To open the **Add Host to Aventail** dialog box, click **Add Host** on the **Extranet Neighborhood Properties | Local** tab.

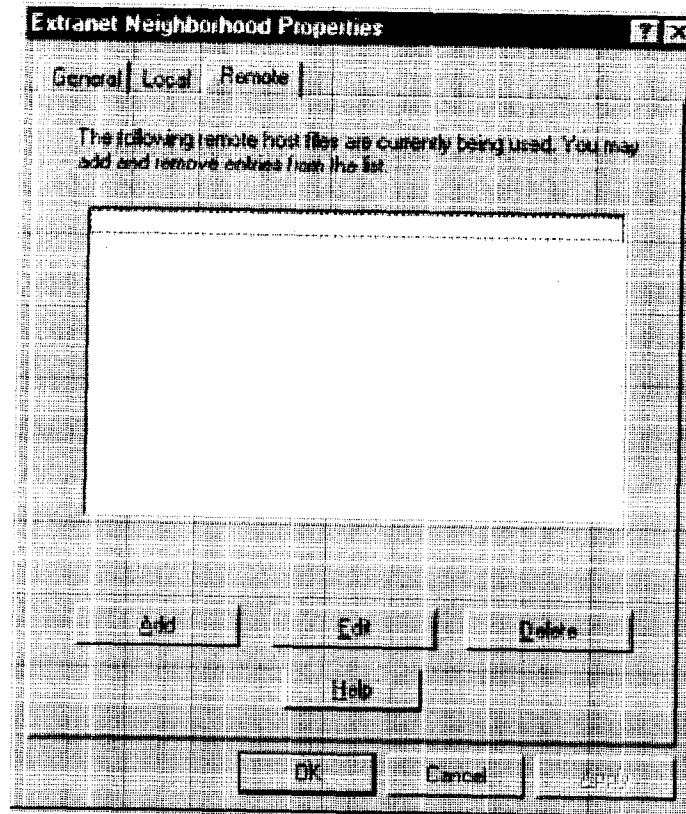
Aventail Connect automatically places hosts within the domain that is selected when you click **Add Host**. Select the correct domain before clicking **Add Host**. You must specify a domain before you can add hosts to that domain.

In the **Host name or IP address** box, be sure to enter the server’s Internet address, not its Windows machine name.

### THE REMOTE TAB

If the network administrator has configured Extranet Neighborhood to use a remote hosts file, this tab displays the information about the currently configured remote hosts file(s). Server name, host name or address, pathname, and user-name are all configurable through the **Remote** tab.





Remote hosts files are always used in conjunction with a local hosts file. When you add a remote hosts file to the list, Extranet Neighborhood adds the path to the local hosts file. Extranet Neighborhood always has a single local hosts file; this file can include references to multiple remote hosts files.

The most common configuration is one remote hosts file (with all domains and hosts in the remote hosts file) and one local hosts file that contains a pointer to the remote hosts file. If you want users to share a common hosts file, and if you want to simplify administration, use a remote hosts file.

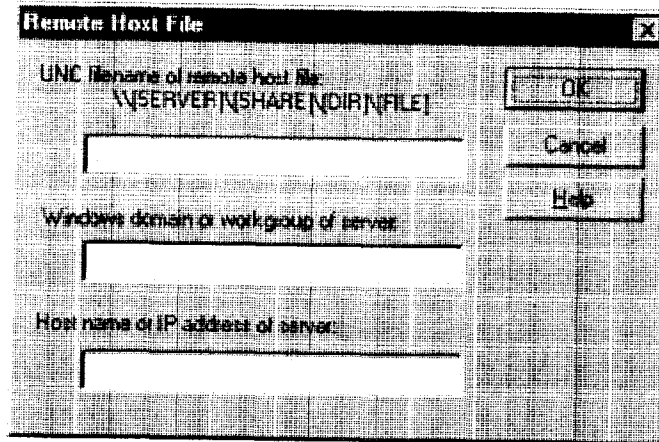
To add entries to the list of remote hosts files, click **Add**. The **Remote Hosts File** dialog box appears, and you can type the names of the remote hosts file(s) you want to add.



**NOTE:** To access remote hosts files, Aventail Connect must be running and configured correctly.

### **Remote Hosts File Dialog Box**

To open the **Remote Hosts File** dialog box, click **Add** on the **Remote** tab.



When entering the Universal Naming Convention (UNC) filename of the remote hosts file that you are adding, note that the [SERVER] name is the Windows machine name, not its IP address or hostname.

In the **Host name or IP address of Server** box, be sure to enter the server's Internet address, not its Windows machine name.



**NOTE:** *Extranet Neighborhood ignores any remote hosts files that it cannot access.*

# Troubleshooting

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

## AVENTAIL CONNECT INSTALLATION PROBLEMS

When the instructions in "Installing" in the *Administrator's Guide* are followed, Aventail Connect installation problems rarely occur. When they do occur, they are often the result of:

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

If any of these are running during a failed installation, close them, uninstall Aventail Connect, reboot, and then re-install Aventail Connect, ensuring that the toolbars, virus-checking utilities, or applications are not automatically restarted when the system reboots.

- **Insufficient RAM or free space on the volume to which Aventail Connect is being installed**

If you suspect either of these as the cause of a failed installation, increase the available resources and retry the installation.

- **Corrupted Aventail Connect installation media, or corrupted or incomplete FTP of Aventail Connect self-extracting, executable installation file**

If you suspect corrupted Aventail Connect installation diskettes as the cause of a failed installation, contact Aventail Technical Support (206.215.0078) for assistance in determining whether the files on the diskettes may have been corrupted and whether Aventail or your vendor must supply replacement diskettes.

If you suspect a corrupted or incomplete FTP transfer of Aventail Connect installation files obtained over the Internet, 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 Aventail Connect was running or from which a previous version of Aventail Connect was not completely uninstalled**

If you suspect either of these circumstances as the cause of a failed installation, contact Aventail Technical Support.

- **Installation script errors**

Aventail Connect 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 Aventail Connect can successfully redirect WinSock application connections:

1. The workstation on which Aventail Connect 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 Aventail Connect 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 Aventail Connect is installed and the extranet (SOCKS) server(s) to which it is configured to redirect connections.

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

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

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

## AVENTAIL CONNECT CONFIGURATION PROBLEMS

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

It is easiest to troubleshoot Aventail Connect configuration problems by creating and testing simple Aventail Connect configuration files, such as those that may be created with the Aventail Connect configuration wizard. However, all references to host and domain names must 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 extranet (SOCKS) server(s) to which Aventail Connect must connect must be known before troubleshooting Aventail Connect configuration problems. Neither Aventail Connect, nor Aventail Technical Support, can discover the IP address or port number of the extranet server(s).*

When troubleshooting Aventail Connect configuration problems, confirm that the Aventail Connect configuration file that is currently selected in the **Configuration File** dialog box is the one intended for testing.

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

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

Click the **Servers** tab, select the server alias and then click **Edit....** Compare the IP address in the **Hostname** or **IP** box with that of the extranet server.

If the extranet server is a SOCKS v5 server, click **SOCKS v4** in the "SOCKS Version" area of the **Servers** tab. Then click **Detect Version**. The selection will revert to **SOCKS v5**, indicating that Aventail Connect detected a SOCKS v5 server running at the IP address specified in the **Hostname** or **IP** box.

If, on the other hand, the extranet server is a SOCKS v4 server, click **SOCKS v5** in the "SOCKS Version" area. Then click **Detect Version**. The selection will revert **SOCKS v4**, indicating that Aventail Connect detected a SOCKS v4 server running at the IP address specified in the **Hostname** or **IP** box.

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

2. Confirm that all Aventail Connect authentication modules are enabled.

Click 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 Aventail Connect to attempt any form of authentication demanded by the extranet server or null (no) authentication. Note the form of authentication demanded by the extranet server and, if necessary, obtain the proper authentication credentials, such as an extranet server username and password, from the extranet server administrator.

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

Click the **Destinations** tab, select the destination that includes the network host to which the extranet server is expected to proxy connections, and then click **Edit....** Confirm that the definition of the Destination includes the network host.

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

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

## APPLICATION AND TCP/IP STACK INTEROPERABILITY PROBLEMS

Aventail Connect is intended to "automatically socksify" all "well-behaved" WinSock applications. Occasionally, you may find WinSock applications that Aventail Connect does not socksify, due to interoperability problems with the application.

Aventail Connect is also intended to run on all WinSock-compliant Microsoft Windows TCP/IP stacks. Aventail Connect does not alter or replace WinSock or any other core TCP/IP components (files) provided by the operating system. Occasionally, you may find WinSock stacks on which Aventail Connect does not run as expected, due to interoperability problems with the stack.

If you suspect an application or stack interoperability problem, report it to Aventail Technical Support. Aventail will make every reasonable effort to resolve interoperability problems.

## AVENTAIL CONNECT TRACE LOGGING

Aventail Connect includes a Logging Tool for tracing Aventail Connect and WinSock activity. Aventail Connect traces are often useful in troubleshooting Aventail Connect network, extranet server, and WinSock application interoperability problems. Aventail Technical Support engineers may request that you perform a verbose trace, log it to a file, and e-mail it to them as an attachment.

### To run an Aventail Connect trace

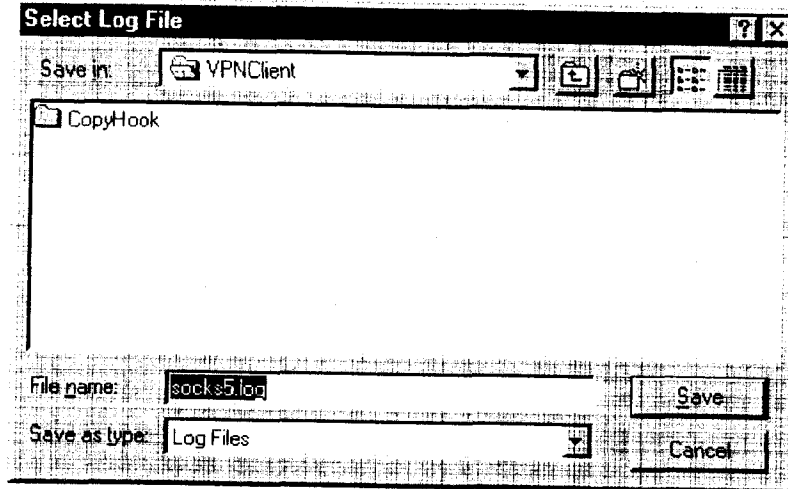
1. Close any WinSock applications that are running on the workstation.
2. If Aventail Connect is running, close it and then restart it.
3. Start an Aventail Connect trace.

In Windows 95, Windows 98, and Windows NT 4.0, right-click the minimized **Aventail Connect** icon in the system tray, and click **Logging Tool**. In Windows 3.1, Windows for Workgroups 3.11, and Windows NT 3.51, double-click the **Logging Tool** icon in the Aventail program group. The Aventail Connect **Logging Tool** window will open, as illustrated in Figure 1, below.

4. On the **Log** menu, confirm that the **Trace** command is checked. If it is not, click **Trace** to enable it.

**To save an Aventail Connect trace to a file**

1. On the **Log** menu, confirm that the **Log To File** command is checked. If it is not, click **Log To File** to enable it.
2. The **Select Log File** dialog box (shown below) appears. Enter a file name and click **Save**.



**ERROR MESSAGES**

Occasionally, you may see an error message while running Aventail Connect. The following table explains some of the more common Aventail Connect error messages.

Error Message:	Meaning
Setup has determined that your computer does not have this support and needs the WinSock 2 patch, available from Microsoft.	SETUP: To install Aventail Connect 3.1, you must first install the Microsoft WinSock 2 upgrade.
The patch is available for download on the Microsoft Web site, at <a href="http://www.microsoft.com/Windows95/downloads/contents/wuadmintools/s_wunetworkingtools/W95Sockets2/default.asp">http://www.microsoft.com/Windows95/downloads/contents/wuadmintools/s_wunetworkingtools/W95Sockets2/default.asp</a> .	SETUP: Location of the Microsoft WinSock 2 upgrade.

Error Message	Meaning
You must have administrator privileges to install.	SETUP: On Windows NT machines, you must have administrative privileges to install or uninstall Aventail Connect.
Setup has detected that a previous installation of (...) is present. Would you like to continue and upgrade to (...)? Pressing NO will leave your existing installation intact and will cause Setup to terminate.	SETUP: Retain the previous installation of Aventail Connect by pressing NO. Replace with the newer installation by pressing YES.
The package does not contain the necessary 3.1 files. Please contact your administrator.	SETUP: Setup cannot find the necessary Aventail Connect 3.1 files.
The package does not contain the necessary 2.6 files. Please contact your administrator.	SETUP: Setup cannot find the necessary Aventail Connect 2.6 files.
The file you have selected is not a valid Aventail setup file. Would you like to create it?	CUSTOMIZER: Create a new setup file, or retain a previous setup file.
Customizer must be run from a valid Customize directory. Your changes will not be saved.	CUSTOMIZER: Must run Customizer from a valid Customize directory.
The Connect executable does not have a valid Aventail digital signature.	The specified signature is not valid.
Connect cannot find your license file, aventail.alf.	Aventail Connect cannot find a valid Aventail license file, aventail.alf.
Connect cannot load because your license file does not contain a license.	The license file exists, but it contains no license.
This version of Connect does not support HTTP servers.	Aventail Connect 2.6 does not support HTTP servers.

## REPORTING AVENTAIL CONNECT PROBLEMS

Report Aventail Connect problems to Aventail Technical Support by completing and submitting an Online Support form on the Support page of the Aventail Web site, <http://www.aventail.com>.



# 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.

**EXTRANET**

A network that is partially accessible to outsiders.

**FIREWALL**

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

**GATEWAY**

A communications device/program that passes data between networks.

**HACKER**

A person who enjoys using computers and has a thorough understanding of how they work, as well as the networks they run on. Often used to mean "cracker," the correct term for someone who accesses computer systems without authorization.

**HOST**

A server connected to the Internet.

**IETF**

Internet Engineering Task Force: An open community of network designers, vendors, etc. who resolve protocol and architectural issues for the quickly evolving 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.

**LAN**

Local area network

**LAYERED SERVICE PROVIDER (LSP)**

A program that is installed just below WinSock 2, allowing two-way communication between the WinSock 2-compatible application and the underlying TCP/IP stack. An LSP can redirect and/or change data before sending the data to the operating system's TCP/IP stack for transport over the network.

**LOG WINDOW**

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

**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 RULES**

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

**ROUTER**

A device that transmits traffic between networks

**SERVER**

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

**SMB**

Server Message Block. A message format used by DOS and Windows for sharing files, directories, and other resources.

**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 and encryption protocol.

**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 its way to the destination machine.

**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.

**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.)

**VIRUS**

A self-replicating code segment that can infect a computer or network, causing minor to major damage

**VPN**

Virtual Private Network: A secure channel used to transmit data over a public network

**WINSOCK**

Windows Sockets. A Windows component that connects a Windows PC to the Internet using TCP/IP.

**WORKSTATION**

Any computer connected to a network.

**X.509**

An ISO format standard for client and server certificates.

**A**

- About command 80
- adding
  - applications to Exclusion/Inclusion List 63
  - destinations 40
  - domains 102, 103
  - hosts 102
  - local domain names 46
  - redirection rules 43
  - remote hosts 104, 105
  - servers 37
- Advanced tab options 62
- alias 36, 41
- applications
  - excluding 63
  - including 63
  - interoperability problems 110
  - securing 62
  - TCP/IP 7, 9, 14
- authentication
  - CHAP 30, 47
  - client 7
  - CRAM 29, 47
  - disabling modules 48
  - enabling modules 48
  - HTTP 30
  - modules 12, 29, 34, 46
  - SOCKS v4 30, 47
  - SSL 29, 47
  - UNPW 30, 47
- Aventail Connect
  - authentication modules 29
  - Config Tool 29, 33, 83
  - configuration files 30, 56
  - configuring 33, 74, 108
  - Customizer 16, 21
  - features 1, 10, 14
  - how does it work? 11
  - in startup directory 16, 28
  - individual installation 16
  - installing 10, 14, 107
  - interface features 14
  - license files 22, 30
  - Logging Tool 29, 83
  - network installation 18
  - overview 7
  - platform requirements 13
  - S5 Ping 29, 83
  - setup 10, 28
  - starting 18

- TCP/IP applications and 9
- tracing activity 29, 85, 110
- v2.5 10
- v3.0 10
- what does it do? 9
- what is it? 7

- Aventail Corporation, about 5
- Aventail Customizer 16, 21, 97, 98
- Aventail ExtraNet Center 95
- Aventail ExtraNet Server 69, 76, 97
- Aventail Knowledge Base 5
- Aventail MultiProxy 68
- Aventail Technical Support 5

**B**

- browsing
  - remote computers 31
  - WINS 99
- browsing mode 96, 97, 102

**C**

- caching 47, 49
- certificate files 28
- certificates
  - chains 52, 59
  - client 7, 28, 55
  - RSA 51
  - server 28, 52
  - validating 53
  - X.509 7, 28
- Certification Authority (CA) 52
- CHAP 30, 47, 50
- ciphers
  - DES 55
  - NULL encryption 55
  - RC4 55
- clearing the log window 91
- client authentication 7
- client certificates 7, 28, 55
- Close command 80
- closing the log window 92
- commands
  - About 80
  - Close 80
  - Configuration File 80
  - Credentials 80
  - Help 80
  - Hide Icon 80
- components, setup package 28
- Config Tool 29, 33, 83, 84
- Configuration File command 80

- configuration files 9, 15, 30, 33, 56
  - password protection 67
- Configuration wizard 18, 33
- configuring
  - Aventail Connect 33, 74, 108
  - CHAP authentication 50
  - CRAM authentication 51
  - Extranet Neighborhood 96
  - hosts files 104
  - HTTP proxies 76
  - MultiProxy 70
  - networks 76
  - SOCKS 4 authentication 48
  - SSL authentication 51
  - UNPW authentication 49
- configuring Extranet Neighborhood 96, 105
- copying
  - log windows 90
- CRAM 29, 47, 51
- creating
  - hosts files 98
  - setup packages 11, 16, 31
- credential cache timeouts 66
- credential caching 47, 49, 66
- credentials 46
  - deleting 82
  - managing 82
- Credentials command 80
- Customizer 16, 21, 97, 98
  - tips 32
- Customizer editor 26
- Customizer options 24
- Customizer wizard 24
- D**
- defining
  - destinations 34
  - hosts 40
  - IP address 40
  - local name resolution 45
  - SOCKS server 35
  - subnets 40
- deleting
  - credential entries 82
- DES 55
- destinations
  - adding 40
  - defining 34
  - editing 42
  - networks 41
  - removing 42
- servers 49
- Diffie-Hellman 55
- directories
  - installation 97
  - startup 16, 28
- distributing
  - configuration files 20
- Domain Name System (DNS) 8, 11
- domains 96, 98, 102, 103, 104
  - names 12, 41, 46
  - strings 11
  - Windows 31
- E**
- editing
  - destinations 42
  - hosts 102
  - redirection rules 44
- enabling password protection 67
- encryption 7, 10, 29, 46, 55
- error messages 111
- example network configuration 76
- excluding applications 63
- Exclusion/Inclusion List
  - adding applications to 63
- Extranet hosts files 31
- Extranet Neighborhood 28, 31
  - browsing mode 96, 97, 102
  - configuring 96, 97, 105
  - how it works 96
  - icon 95, 97, 104
  - installing 97
  - launching 98
  - overview 95
  - properties 101
  - remote access and 95
  - Search feature 97, 102
- Extranet servers 33, 47, 76, 82
- extranet servers 35
- extranets 6, 35
- F**
- file servers 18
- files
  - certificate 28
  - configuration 9, 15, 30, 33, 56
  - hosts 31, 95, 96, 97
  - license 22, 30
  - local hosts 98, 101, 105
  - reloading 104
  - remote hosts 98
  - SEEHosts 98

- shared configuration 19
  - trusted root 28, 53, 55
- filtering messages in log window 88
- firewalls 6, 68
- G**
- Getting Started 6
- Glossary 113
- H**
- Help command 80
- Hide Icon command 80
- hostname 11, 36, 41, 45
- hosts 31
  - adding 102, 104
  - defining 40, 41
  - editing 102
  - local 101, 105
  - remote 8, 104
- hosts files
  - adding 95, 97
  - configuring 104
  - creating 98
  - locking 103
  - populating 97
  - SEEHosts 95
  - unlocking 103
- HTTP authentication 30
- HTTP proxies 68
  - configuring 76
- I**
- icon 95, 97, 104
- including applications 63
- individual installation 16
- installation directory 97
- installation pathname 28
- installing Aventail Connect 10, 14, 107
- installing Extranet Neighborhood 97
- Internet Engineering Task Force (IETF) 6
- Introduction 95
- IP address 8, 11, 36, 40, 41
- K**
- keys
  - pairs 51
  - private 51
  - public 51
- L**
- launching Extranet Neighborhood 98
- Layered Service Provider (LSP) 9
- license files 22, 30
- loading
  - packages 31
  - local hosts files 97, 98, 101, 105
  - local name resolution 34, 45
  - locking hosts files 103
  - log files, saving 87
  - Logging Tool 29, 83, 84
- M**
- managing authentication modules 46
- managing credentials 82
- menu commands 80
- multiple firewall traversal 68
- MultiProxy 68
  - configuring 70
- N**
- NetBIOS 96
- network installation 18
- Network Neighborhood 95, 97
- networks
  - configuring 76
  - connectivity problems 108
  - destinations 41
  - security 6
- O**
- options
  - Customizer 24
- P**
- password protection 67
- pathname, installation 28
- ping 29, 92
- platform requirements 97
- platforms 7, 10, 13, 28
- ports 36
- printing
  - log windows 91
- proxies 6, 44, 72, 77
  - HTTP 68
- proxy chaining 72
- R**
- RC4 55
- redirection rules 11, 15, 34, 40, 42, 96
- reloading hosts files 104
- remote access 95
- remote computers 31
- remote hosts 8
- remote hosts files 98, 104, 105
- removing
  - destinations 42
  - local domain names 46
  - redirection rules 45
- RSA 51

- S**
- S5 Ping 29, 83, 92
  - saving
    - log files 87
    - setup packages 32
  - Search feature 97, 102
  - Secure Extranet Explorer
    - overview 95
    - platform requirements 97
  - Secure Sockets Layer (SSL) 10, 29, 47, 51
  - securing applications 65
  - securing selected applications 62
  - security
    - firewalls 6
    - network 6
    - protocols 6
  - SEEHhosts file 98
  - SEEHhosts files 31
  - server certificates 28, 52
  - servers
    - adding 37
    - alias 36
    - Aventail ExtraNet Server 97
    - destination 49
    - Extranet 33, 47, 76, 82
    - file 18
    - SOCKS 35, 68, 82
    - WINS 31, 96, 97, 103
  - setup 10, 16, 28
  - setup package components 28
  - setup packages 16, 22, 31
  - shared configuration files 19
  - SOCKS 12, 15, 82
  - SOCKS servers 35, 68
  - SOCKS tunneling 62
  - SOCKS v4 30, 47, 48
  - SOCKS v5 6, 7, 38, 46, 92
  - SSL compression 55
  - starting Aventail Connect 18
  - startup directory 16, 28
  - subnets 40, 41
  - system menu commands 80
- T**
- TCP 96
  - TCP/IP
    - applications 7, 9, 14
    - overview 8
    - stack 9, 11, 45, 110
    - WinSock and 7
  - Technical Support 5
- To 64
  - traceroute 29, 92
  - tracing Aventail Connect activity 29, 85, 110
  - Troubleshooting 107
  - trusted root files 28, 53, 55
  - tunneling, SOCKS 62
- U**
- unattended setup mode 28
  - unlocking hosts files 103
  - UNPW 30, 47, 49
  - User Datagram Protocol (UDP) 7
  - utilities
    - Config Tool 29, 83
    - Logging Tool 29, 83
    - ping 29
    - S5 Ping 29, 83
    - traceroute 29
- W**
- Web browsers
    - HTTP proxies and 72, 74
  - Windows 95
    - WinSock and 10, 11, 13
  - Windows Explorer 95
  - WINS browsing 99
  - WINS servers 31, 96, 103
  - WinSock 7, 10, 11
- X**
- X.509 certificates 7, 28



# EX. 1051

For Declaration of Chris Hopen

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In the Reexamination of: )  
Edmund Munger, et al. )  
 )  
U.S. Patent No.: 6,502,135 )  
Filed: February 15, 2000 ) Examiner:  
Issued: December 31, 2002 ) Andrew L. Nalven  
 )  
For: AGILE NETWORK PROTOCOL ) Group Art Unit: 3992  
FOR SECURE COMMUNICATIONS )  
WITH ASSURED SYSTEM )  
AVAILABILITY )  
 )  
Reexamination Proceeding )  
Control No.: 95/001,269 )  
Filed: December 8, 2009 )

Declaration of Jason Nieh, Ph.D., Pursuant to 37 C.F.R. § 1.132

Pursuant to 37 C.F.R. § 1.132, I declare that the following statements are true to the best of my knowledge, information, and belief, formed after reasonable inquiry under the circumstances.

**Background**

1. I have over 15 years of experience with operating systems and distributed systems. More specifically, my experience includes remote access, computer networking, and computer security. Examples of my experience are evidenced by my publication of papers in top-tier networking and security conferences, service on programming committees for networking and security conferences, awards for research work, and receipt of research grants in the field of networking and security. My qualifications, including a description of all of this information, may be found in my curriculum vitae, which is attached hereto as Exhibit A.

2. I earned a Bachelor of Science degree from the Massachusetts Institute of Technology in Electrical Engineering in 1989. I earned a Masters of Science degree from Stanford University in Electrical Engineering in 1990. I also received my Ph.D. in Electrical Engineering from Stanford University in 1999.

**EXHIBIT A-3**

Control No.: 95/001,269  
Declaration of Jason Nieh, Ph.D.

3. I joined Columbia University as a faculty member in 1999, where I am now a tenured Associate Professor in the Department of Computer Science. I am also currently the director of the Network Computer Laboratory at Columbia University.

4. My research interests include mobile computing, operating systems, distributed systems, thin-client computing, web and multimedia systems, and performance evaluation. I have supervised a number of Ph.D. students who worked on and completed dissertations in the area of networking and security. I also teach courses in advanced operating systems and mobile computing, both of which involve computer networking and security.

5. I have also served as an expert in various litigations in the fields of computer networking and security, which include virtual private networking.

#### **Resources I have Consulted**

6. I have been retained by the Patent Owner, VirnetX, Inc., to offer my opinion of the patentability of claims 1, 3, 4, 6-10, and 12 of U.S. Patent Number 6,502,135 (“the ‘135 Patent”) in view of the Office Action dated January 15, 2010 (“the Office Action”) received by the Patent Owner in the reexamination of the ‘135 Patent.

7. In preparing this declaration, I have reviewed the ‘135 Patent, including the claims. I have also reviewed the outstanding Office Action. I have also reviewed the Request for *Inter Partes* Reexamination of Patent (“the Request”) to the extent it is adopted by the Office Action. I have also reviewed Appendix A to the Request (“Appendix A”) to the extent that it is adopted in the Office Action. Lastly, I have reviewed Aventail Connect v3.1/v2.6 Administrator’s Guide (“Aventail”), the reference upon which the rejection in the Office Action is based.

8. A detailed explanation of the basis for my opinions is set forth in the remainder of this declaration.

#### **Detailed Basis for My Opinion**

I provide here a brief description of the system disclosed in Aventail.

9. As I stated above, I have read the ‘135 Patent, including the claims, and understand independent claim 1 to recite “[a] method of transparently creating a virtual private network (VPN) between a client computer and a target computer, comprising the steps of: (1) generating from the client computer a Domain Name Service (DNS) request that requests an IP address corresponding to a domain name associated with the target computer; (2) determining whether

Control No.: 95/001,269  
Declaration of Jason Nieh, Ph.D.

the DNS request transmitted in step (1) is requesting access to a secure web site; and (3) in response to determining that the DNS request in step (2) is requesting access to a secure target web site, automatically initiating the VPN between the client computer and the target computer.”

10. Similarly, I understand independent claim 10 to recite “[a] system that transparently creates a virtual private network (VPN) between a client computer and a secure target computer, comprising: a DNS proxy server that receives a request from the client computer to look up an IP address for a domain name, wherein the DNS proxy server returns the IP address for the requested domain name if it is determined that access to a non-secure web site has been requested, and wherein the DNS proxy server generates a request to create the VPN between the client computer and the secure target computer if it is determined that access to a secure web site has been requested; and a gatekeeper computer that allocates resources for the VPN between the client computer and the secure web computer in response to the request by the DNS proxy server.”

11. After reviewing the Aventail reference, I understand Aventail to disclose a system for transmitting data between two computers using the SOCKS protocol. The system according to Aventail routes certain, predefined network traffic from a WinSock (Windows sockets) application to an extranet (SOCKS) server, possibly through successive servers. Upon receipt of the network traffic, the SOCKS server then transmits the network traffic to the Internet or external network. Aventail’s disclosure is limited to connections created at the socket layer of the network architecture.

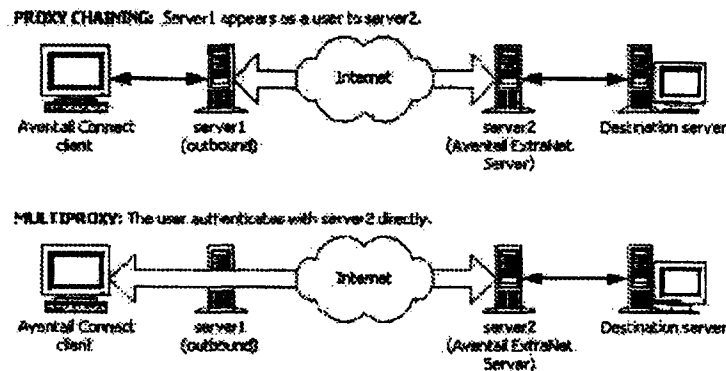
12. I note that pages 9-12 of Aventail discuss the basics of the operation of Aventail Connect, the software necessary to implement the system disclosed in Aventail. According to page 9 of Aventail, a component of the Aventail Connect software described in the reference resides between WinSock and the underlying TCP/IP stack. Accordingly, Aventail Connect is able to intercept all connection requests from the user, and determines whether each request matches local, preset criteria for redirection to a SOCKS server.

13. According to page 12 of Aventail, if redirection is appropriate, then Aventail Connect creates a false DNS entry to return to the requesting application. Aventail discloses that Aventail Connect then forwards the destination hostname identified in the DNS request to the extranet SOCK server over a SOCKS connection.

Control No.: 95/001,269  
Declaration of Jason Nieh, Ph.D.

14. Although Aventail is generally silent on the operation of the SOCKS server, I understand from page 12 that the SOCKS server performs the hostname resolution. Once the hostname is resolved, the user can transmit data over a SOCKS connection to the SOCKS server. The SOCKS server, then, separately relays that transmitted data to the target.

15. Page 12 of the Request also cites to the “Proxy Chaining” and “MultiProxy” modes disclosed in Aventail at pages 68-73. I have reproduced below a figure taken from page 72 of Aventail depicting these two modes.



16. In the “Proxy Chaining” mode, Aventail indicates that a user can communicate with a target via a number of proxies such that each proxy server acts as a client to the next downstream proxy server. As shown above, in this mode, the user does not communicate directly with the proxy servers other than the one immediately downstream from it.

17. In the “MultiProxy” mode, Aventail indicates that the user, via Aventail Connect, authenticates with each successive proxy server directly.

18. Regardless of whether one of these modes is enabled, as shown in the figure, an external SOCKS server is necessary and the operation of Aventail Connect, for the purposes of my opinion, does not materially differ based on whether one of these modes is enabled.

**Aventail has not been shown to disclose a virtual private network according to claim 1.**

19. Aventail has not been shown to disclose the VPN claimed in claim 1 of the ‘135 Patent for at least three reasons.

20. First, 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

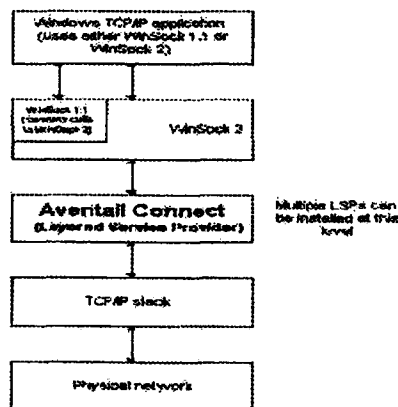
Control No.: 95/001,269  
Declaration of Jason Nieh, Ph.D.

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.

21. For example, suppose two computers, A and B, reside on a public network. Further, suppose two computers, X and Y, reside on a private network. If A establishes a VPN connection with X and Y's network to address data to X, and B separately establishes a VPN connection with X and Y's network to address data to Y, then A would nevertheless be able to address data to B, X, and Y without additional set up. This is true because A, B, X, and Y would all be a part of the same VPN.

22. In contrast, suppose, according to Aventail, which only discloses communications at the socket layer, A establishes a SOCKS connection with a SOCKS server for relaying data to X, and B separately establishes a SOCKS connection with the SOCKS server for relaying data to Y. In this situation, not only would A be unable to address data to Y without establishing a separate SOCKS connection (the alleged VPN according to the Office Action), but A would be unable to address data to B over the secure connection. This is one example of how the cited portions of Aventail fail to disclose a VPN.

23. Second, according to Aventail, Aventail Connect's fundamental operation is incompatible with users attempting to transmit data that is sensitive to network information. As I stated above, Aventail discloses that Aventail Connect operates between the WinSock and TCP/IP layers. The figure I have reproduced below from page 9 of Aventail depicts this operation.



Control No.: 95/001,269  
Declaration of Jason Nieh, Ph.D.

24. Because Aventail discloses that Aventail Connect operates between these layers, Aventail Connect can intercept DNS requests requested by the user. Aventail discloses that 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.

26. Third, Aventail has not been shown to disclose a VPN because computers connected according to Aventail do not communicate directly with each other. Aventail discloses a system where a client on a public network transmits data to a SOCKS server via a singular, point-to-point SOCKS connection at the socket layer of the network architecture. The SOCKS server then relays that data to a target computer on a private network on which the SOCKS server also resides. All communications between the client and target stop and start at the intermediate SOCKS server. The client cannot open a connection with the target itself. Therefore, one skilled in the art would not have considered the client and target to be virtually on the same private network. Instead, the client computer and target computer would have been understood to be deliberately separated by the intermediate SOCKS server.

27. For the reasons stated above, I do not believe that Aventail has been shown to teach or disclose the “VPN” recited in claim 1. Because claims 2, 4, and 6-9 depend from claim 1, I also do not believe that Aventail has been shown to teach or disclose the inventions claimed in claims 2, 4, and 6-9.

Aventail has not been shown to disclose a virtual private network according to claim 10.

28. As I stated above, independent claim 10 similarly recites a “VPN between a client computer and the secure target computer.” For at least the reasons I have stated above, I do not believe that Aventail has been shown to teach or disclose the invention recited in claim 10.

29. Because claim 12 depends from claim 10, I also do not believe that Aventail has been shown to teach or disclose the invention claimed in claim 12.

Aventail has not been shown to teach a DNS proxy server according to claim 10.

30. As I stated above, claim 10 recites a “DNS proxy server” that 1) “returns the IP address for the requested domain name if it is determined that access to a non-secure web site has been requested” and that 2) also “generates a request to create the VPN . . . if it is determined that access to a secure web site has been requested.”

31. The Office Action and Request allege that Aventail Connect is the claimed DNS proxy server.

32. As I have stated previously, Aventail discloses that Aventail Connect intercept all DNS requests. According to Aventail, at page 11, “[i]f 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.” Thus, Aventail discloses that Aventail Connect does not return the IP address if the DNS request requests the address for a non-secure web site. As such, Aventail Connect does not correspond to the DNS proxy server recited in claim 10.

33. For at least this reason, I do not believe that Aventail has been shown to teach or disclose the invention recited in claim 10. Because claim 12 depends from claim 10, I also do not believe that Aventail has been shown to teach or disclose the invention claimed in claim 12.

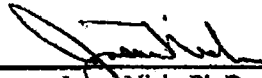
**Truth and Accuracy of Statements**

34. I further 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 willful false statements or the like may jeopardize the validity of the application or any patent issuing thereon.



Control No.: 95/001,269  
Declaration of Jason Nieh, Ph.D.

Signed at New York, New York this 15th day of April, 2010.

  
\_\_\_\_\_  
Jason Nieh, Ph.D.

WDC99 1837192-5.077580.0089

# EX. 1052

For Declaration of Chris Hopen

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re <i>Inter Partes</i> Reexamination of:	)	
	)	
Edmund Munger et al.	)	Control No.: 95/001,682
	)	
U.S. Patent No. 6,502,135	)	Group Art Unit: 3992
	)	
Issued: December 31, 2002	)	Examiner: Behzad Peikari
	)	
For: AGILE NETWORK PROTOCOL FOR SECURE	)	Confirmation No.: 1074
COMMUNICATIONS WITH ASSURED	)	
SYSTEM AVAILABILITY	)	

Mail Stop *Inter Partes* Reexam  
Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

**Declaration of Angelos D. Keromytis, Ph.D.**

I declare that the following statements are true to the best of my knowledge, information, and belief, formed after reasonable inquiry under the circumstances.

I, ANGELOS D. KEROMYTIS, declare as follows:

1. I have been retained by VirnetX Inc. (“VirnetX”) for the above-referenced reexamination proceeding. I understand that this reexamination involves U.S. Patent No. 6,502,135 (“the ‘135 patent”). I further understand that the ‘135 patent is assigned to VirnetX and that it is part of a family of patents (“Munger patent family”) that stems from U.S. provisional application nos. 60/106,261 (“the ‘261 application”), filed on October 30, 1998, and 60/137,704 (“the ‘704 application”), filed on June 7, 1999. I also understand that the ‘135 patent is a continuation-in-part of U.S. application no. 09/429,643 (now U.S. Patent No. 7,010,604), which claims priority to the ‘261 and ‘704 applications.

**I. RESOURCES I HAVE CONSULTED**

2. I have reviewed the ‘135 patent, including claims 1-18. I have also reviewed a Request for *Inter Partes* Reexamination of the ‘135 patent filed by Apple Inc. with the U.S. Patent

and Trademark Office on July 11, 2011 (“Request” or “Req.”), as well as its accompanying exhibits.<sup>1</sup> Additionally, I have reviewed an Order Granting Request for *Inter Partes* Reexamination of the ‘135 patent (“the Order”) mailed on October 3, 2011, and an Office Action (“the Office Action”) mailed on February 15, 2012.<sup>2</sup>

3. I have also studied the following documents cited in and included with the Request and/or Office Action: Aventail Connect v3.1/2.6 Administrator’s Guide (Req. Ex. X1) (hereinafter “*Aventail v3.1*”); Aventail Connect v3.01/2.51 Administrator’s Guide (Req. Ex. X2) (hereinafter “*Aventail v3.01*”); AutoSOCKS v2.1 Administrator’s Guide (Req. Ex. X3) (hereinafter “*AutoSOCKS*”); Wang, Broadband Forum TR-025: Core Network Architecture Recommendations For Access to Legacy Data Networks over ADSL, Issue 1.0 (“*Wang*”); U.S. Patent Number 6,496,867 (“*Beser*”); Kent, “Security Architecture for IP,” RFC 2401 (“*Kent*”); Reed, “Proxies for Anonymous Routing”, 12th Annual Computer Security Applications Conference (“*Reed*”); *BinGO!* User’s Guide and *BinGO!* Extended Feature Reference (“*BinGO*”); U.S. Patent Number 6,615,357 (“*Boden*”); U.S. Patent Number 6,182,141 (“*Blum*”); U.S. Patent Number 4,885,778 (“*Weiss*”); Goldschlag et al., “Hiding Routing Information,” (“*Goldschlag*”); Ferguson et al., “What Is a VPN,” (“*Ferguson*”); RFC 1034, “Domain Names—Concepts and Facilities” (“RFC 1034”); RFC 1035, “Domain Names—Implementation and Specification” (“RFC 1035”); RFC 1123, “Requirements for Internet Hosts—Applications and Support” (“RFC 1123”); RFC 2068, “Hypertext Transfer Protocol – HTTP/1.1” (“RFC 2068”); RFC 1928, “Socks Protocol Version 5” (“RFC 1928”); RFC 1180, “A TCP/IP Tutorial” (“RFC 1180”); RFC 1661, “The Point-to-Point Protocol (PPP)” (“RFC 1661”); RFC 1968, “The PPP Encryption Control Protocol (ECP)” (“RFC 1968”); RFC 2420, “The PPP Triple-DES Encryption Protocol (3DESE)” (“RFC 2420”); RFC 2661, “Layer Two Tunneling Protocol ‘L2TP’” (“RFC 2661”); RFC 2118, “Microsoft Point-To-Point Encryption (MPPE) Protocol” (“RFC 2118”); RFC 2364, “PPP Over AAL5” (“RFC 2364”); RFC 2663, “IP Network Address Translator (NAT) Terminology and Considerations” (“RFC 2663”); and RFC 1483,

---

<sup>1</sup> I refer to the Request for *Inter Partes* Reexamination as “the Request” and, correspondingly, I will refer to Apple Inc. as “the Requester.”

<sup>2</sup> The Office Action incorporates nearly all of the Request by reference. For that reason, when I sometimes refer to “the Request,” I am also referring to the Office Action.

“Multiprotocol Encapsulation over ATM Adaption Layer 5” (“RFC 1483”).<sup>3</sup>

4. I am familiar with the level of ordinary skill in the art with respect to the inventions of the '135 patent as of February 15, 2000, when the application for the '135 patent was filed. Specifically, based on my review of the technology, the educational level of active workers in the field, and drawing on my own experience, I believe a person of ordinary skill in art at that time would have had a master's degree in computer science or computer engineering, as well as two years of experience in computer networking with some accompanying exposure to network security.

5. I have been asked to consider how one of ordinary skill in the art would have understood the references mentioned above. My findings are set forth below.

## II. QUALIFICATIONS

6. I have a great deal of experience and familiarity with computer and network security, and have been working in this field since 1993.

7. I am currently an Associate Professor of Computer Science at Columbia University, as well as Director of the University's Network Security Laboratory. I joined Columbia in 2001 as an Assistant Professor, after receiving my M.Sc. and Ph.D. degrees in Computer Science, both from the University of Pennsylvania. My Ph.D. dissertation work was on the topic of secure access control for distributed systems and, in particular, on the management of trust in distributed computer networks.

8. I received my B.Sc. in Computer Science from the University of Crete, in Greece, in 1996. During my undergraduate studies, I worked as system administrator in the Computing Center at the University of Crete. Following that, I worked as network engineer at the first commercial Internet Service Provider (“ISP”) in Greece, FORTHnet SA, where I was exposed to many network security issues.

9. I have actively participated in the Internet Engineering Task Force (“IETF”), a standards-setting body for the Internet, since 1995. In the late 1990s and early 2000s, my work with the IETF was primarily within the Internet Protocol Security (“IPsec”) Working Group. In addition to contributing to the specification of the IPsec standards, I wrote the first implementation of the

---

<sup>3</sup> Although I listed dates in these citations, I am not testifying to whether any of these references were actually publicly distributed on the date listed.

Photuris key management protocol (now RFC 2522). I also contributed to the first open-source implementation of the IKSAMP/IKE key management protocol for the open-source BSD operating system (now RFC 2409), and developed the first such implementation for the Linux operating system. My Linux implementation, named Pluto, was adopted by the National Institute of Standards and Technology (“NIST”) in 1999. In addition, my implementation of IPsec for the open-source BSD operating system is currently used by many companies and governments around the world, and serves as the basis for several commercial products that employ cryptographic communications. In 1999, I architected and implemented the first open-source framework for supporting hardware cryptographic accelerators. This framework is used in the open-source OpenBSD, NetBSD, FreeBSD, and Linux operating systems. My work in implementing firewalls and other cryptographic and network protocols has resulted in commercial systems and publications in refereed technical conferences and academic journals. I served as Working Group Secretary for the IETF IPsec Working Group (2003-2005) and as Security Area Advisor to the IETF at large (2003-2008).

10. In my current position at Columbia University, I work with a large group of graduate and postgraduate students in the area of cybersecurity. My past students now work in this field as university professors, as technical researchers for research laboratories, or as engineers for telecommunications companies. I have received federal, state, and corporate sponsorship to conduct cybersecurity research from the Department of Defense, the National Security Agency, the Defense Advanced Research Projects Agency (“DARPA”), the National Science Foundation, the Department of Homeland Security, the Air Force, the Office for Naval Research, the Army Research Office, the Department of the Interior, the National Reconnaissance Office, New York State, Google, Intel, Cisco, and others. In my ten years as a professor, I have received over 36 million dollars to support my research in cybersecurity. I also regularly teach courses on cybersecurity, in addition to more general courses in computer science.

11. I have published over 200 technical papers in refereed journals, conferences, and workshops, all of which are directed to various areas of cybersecurity. I have also authored a book, coauthored another book, and contributed chapters for many other books that relate to cybersecurity. Between 1999 and 2010, I have drafted or codrafted eight standards documents that were published as Request for Comments (“RFCs”). Several of these RFCs are directly related to IP security. For example, RFC 6042 relates to transport layer security; RFC 5708, RFC 2792, and RFC 2704 relate to key signature and encoding for trust management; and RFC 3586 relates to IP security policy requirements. Additionally, I am a coinventor on twelve issued U.S. patents, and have several other

applications pending. Most of these patents and pending applications are related to network and systems security.

12. I have chaired several international technical conferences and workshops in cybersecurity, including, for example, the International Conference on Financial Cryptography and Data Security (FC), ACM Computer and Communication Security (CCS), and the New Security Paradigms Workshop (NSPW). I have also served in over eighty technical program committees for such events. From 2004-2010, I served as Associate Editor for the premier technical journal on cybersecurity—the ACM Transactions on Information and Systems Security (TISSEC). Additionally, I have served on several advisory workshops to the United States Government on cybersecurity, including, among others, the Office of the Director of National Intelligence (ODNI)/National Security Agency (NSA) Invitational Workshop on Computational Cybersecurity in Compromised Environments (C3E) (2011), the Office of Naval Research (ONR) Workshop on Host Computer Security (2010), the Intelligence Community Technical Exchange on Moving Target (2010), Lockheed Martin Future Security Threats Workshop (2009), and the ARO/FSTC Workshop on Insider Attack and Cyber Security.

13. In addition to this work, I have cofounded two companies in cybersecurity. One company, StackSafe Inc. (formerly Revive Systems Inc.), was a provider of a virtualized reproduction staging environment that includes automated testing, analysis, and reporting for IT operations teams. I was with this company from its founding in 2005 until 2009. The second company, Allure Security Technologies (founded in 2010), develops deception-based solutions for detecting and mitigating the malicious cyber-insider threat, commercializing technology developed at Columbia through DHS and DARPA grants and a DARPA SBIR contract.

14. My curriculum vitae, which is appended to this declaration, details my background and technical qualifications. Although I am being compensated at my standard rate of \$500/hour for my work on this declaration, the compensation in no way affects the statements in this declaration.

### **III. BACKGROUND OF THE '135 PATENT**

15. Before turning to a discussion of the references relied on in the Request and the Office Action, I summarize my understanding of certain embodiments disclosed in the '135 patent. Generally speaking, the '135 patent discloses embodiments relating to establishing virtual private networks (VPNs) and/or virtual private links between devices connected to a network. For example, certain embodiments of the '135 patent may establish a VPN between a client computer and a target

computer in response to a request received from the client computer for a network address corresponding to a domain name associated with the target computer. ('135 patent 37:63-39:41.)

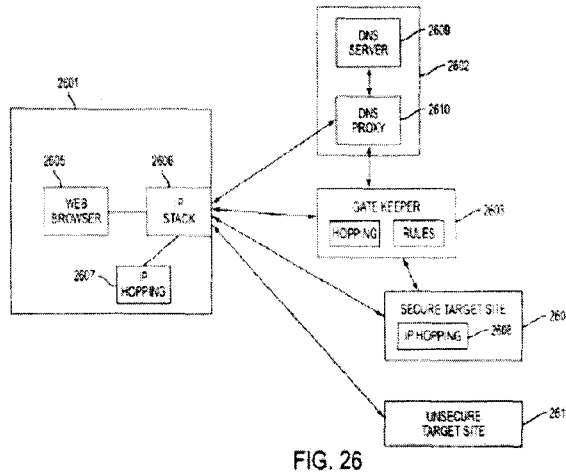


FIG. 26

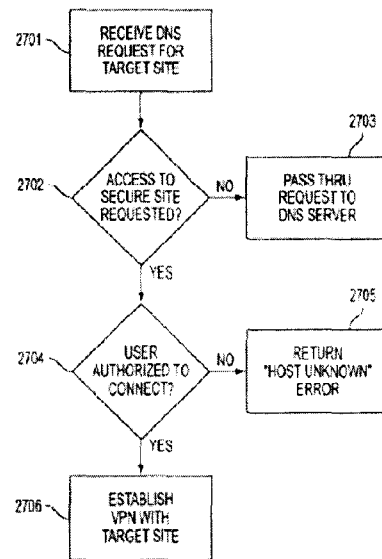


FIG. 27

16. As shown in Figures 26 and 27 of the '135 patent, reproduced above, a computer 2601 may generate a domain name service (DNS) request for an internet protocol (IP) address corresponding to a domain name of a target computer, such as secure target site 2604 and/or unsecure target site 2611. DNS proxy server 2610 may receive the DNS request from computer 2601 and determine whether the DNS request is requesting access to a secure web site. ('135 patent 38:23-30, 39:2:6.)

17. If the DNS request is requesting access to a secure web site, DNS proxy server 2610 may determine whether the user and/or computer 2601 is authorized to access the secure web site. ('135 patent 38:25-30, 39:7-20.) If so, a VPN may be automatically initiated between computer 2601 and secure target site 2604. ('135 patent 38:30-43, 39:22-33.) In certain embodiments, this may include DNS proxy 2610 sending a request to create the VPN to a gatekeeper 2603; gatekeeper 2603 may allocate resources for the VPN in response to the request. ('135 patent 38:30-43.) The '135 patent makes clear that the gatekeeper 2603 may be implemented separately from, or as a part of, modified DNS server 2602 that includes DNS proxy 2610. ('135 patent 38:53-60.)

18. If, on the other hand, the DNS request is requesting access to a non-secure website, DNS proxy server 2610 may pass the request through to conventional DNS server 2609, which may return the IP address of unsecure target website 2611. ('135 patent 38:43-52.)

19. The claims of the '135 patent are directed to some of these embodiments.



#### IV. REFERENCES CITED

##### A. *Aventail*

20. *Aventail v3.1* is an administrator's guide for configuring Aventail Connect, a client component of the Aventail ExtraNet Center, an extranet solution. (*Aventail v3.1* 3, 7.) Aventail Connect works in connection with extranet servers running the SOCKS protocol, including the Aventail ExtraNet Server, the SOCKS 5 server component of the Aventail ExtraNet Server. (*Id.* at 7.)

21. *Aventail v3.1* discloses two primary embodiments:

- (1) Aventail Connect may be used to provide secure inbound access, *i.e.*, allowing an organization to provide its mobile employees and partners secure access to the organization's private network, extranet, or LAN from remote locations over the Internet. (*E.g., id.* at 5, 7, 77.)
- (2) Aventail Connect may also be used as a simple proxy client for managed outbound access, *e.g.*, from a corporate network to the Internet, through a SOCKS compliant server. (*E.g., id.* at 5, 7, 68.)

22. In the first embodiment, Aventail Connect accesses the private network through the Aventail ExtraNet Server. (*Id.* at 77.) The Aventail ExtraNet Server restricts inbound access by allowing only authorized client computers running Aventail Connect to send or receive data to a computer on the private network, and provides an encrypted connection between the Aventail ExtraNet Server and the external client computer. (*See, e.g., id.* at 72).

23. In the second embodiment, Aventail Connect may be configured to route certain traffic from a client computer running Aventail Connect to a SOCKS compliant proxy server to traverse a firewall and access a remote host beyond the firewall. (*See id.* at 6-7.) In some cases, multiple firewalls may be traversed using successive proxy servers (*id.* at 68-73). Routing is accomplished, in part, by an administrator first defining what SOCKS proxy servers that Aventail Connect may use when routing connections. (*Id.* at 35-37.) Any SOCKS compliant proxy server may be used. (*See id.* at 37 (figure depicting that a user may choose SOCKS v4, v5, or HTTP proxy).) The administrator then defines destinations (*e.g.*, hostnames) that may be routed through the previously defined servers. (*Id.* at 39.) After the destinations have been configured, the administrator may create redirection rules. A redirection rule defines, for a defined destination, what type of traffic (*i.e.*, TCP and/or UDP) will be allowed to be routed to that destination, and which proxy server will be used to route that traffic. (*Id.* at 42-44.) The redirection rules may be arranged to prioritize how a destination will be handled. (*Id.* at 43.)

24. Based on my review, *Aventail v3.01* and *AutoSOCKS* incorporates similar subject matter of, and is substantially similar to *Aventail v3.1*. Moreover, it is my understanding that many, if not all, of the allegations made by the Request and adopted by the Office Action with regard to *Aventail v3.01* and *AutoSOCKS* are substantially similar to the allegations made with respect to *Aventail v3.1*. Therefore, for the purpose of this declaration I will refer to all three references collectively as “*Aventail*.” All citations will be based on *Aventail v3.1*, but I will also refer generally to those sections of *Aventail v3.01* and *AutoSOCKS* that disclose similar subject matter.

25. I understand independent claim 1 to recite, among other things, “(3) in response to determining that the DNS request in step (2) is requesting access to a secure target web site, automatically initiating the VPN between the client computer and the target computer.”

26. I have been asked to assume that *Aventail* teaches a VPN, generally. Even if *Aventail* is viewed as showing a VPN, however, it is my opinion that *Aventail* still does not contain all of the features of claim 1. For example, in my opinion, *Aventail* does not disclose at least the feature of initiating a VPN in response to determining that a DNS request is requesting access to a secure target web site, as recited by step (3) of claim 1.

27. The Request contends that if a hostname matches a redirection rule then a connection would be established between a client computer running *Aventail Connect* and the *Aventail Extranet Server*, and, if authentication was successful, the purported VPN would be established. (Req. at 43 (citing *Aventail v3.1* 12.)

28. The Request does not say how a DNS request is determined to be requesting access to a secure web site in *Aventail*. Rather, the Request points to two different things: (1) evaluating a hostname, and (2) evaluating a connection request. (Req. at 41) Thus, the Request is unclear as to what determines that a DNS request is requesting access to a secure target web site.

29. If the Request is contending that evaluating a hostname against a redirection rule is a determination step then the only thing *Aventail* discloses in response to that determination is the creation of a false DNS request. Moreover, *Aventail* does not disclose that the false DNS entry initiates a connection request. *Aventail* teaches on page 12 that, in step 2, *Aventail Connect* merely “checks” an already existing connection request to determine whether the request contains a false DNS entry. Whether a completed connection is subsequently encrypted or not is not disclosed by *Aventail* as having anything to do with a DNS request, let alone “in response to determining that the DNS request in step (2) is requesting access to a secure target web site,” as recited in claim 1.

30. Similarly, in my opinion, evaluating the connection request for the presence of a false DNS entry in *Aventail* does not involve determining that a DNS request is requesting access to a secure target web site. A false DNS entry may be created irrespective of whether a target web site is determined to be secure or not. For example, a false DNS entry will be created as a result of selecting a DNS proxy option, *i.e.*, to proxy all DNS lookups that cannot be looked up directly, whether for secure destinations or not. (*See Aventail v3.1* 12 (step 1, bullet point 3).) Moreover, a “false DNS entry” is not a DNS request.

31. *Aventail* does not disclose that encryption (*i.e.*, the purported VPN) is automatically initiated in response to determining that the DNS request in step (2) is requesting access to a secure target web site, as recited by claim 1. *Aventail* explains that encryption is initiated, if at all, “[w]hen the connection is completed” to the SOCKS server. (*Aventail v3.1* 12 (step 2.b).) But *Aventail* does not teach any link between a *DNS request* and the encryption, much less that it is automatically initiated in response to determining that the DNS request in step (2) is requesting access to a secure target web site, as recited by claim 1.

32. Furthermore, the Request contends that *Aventail* discloses the step of generating a DNS request that requests an IP address because *Aventail Connect* “automatically routes appropriate network traffic,” and points to three different methods of how “*Aventail Connect* . . . resolve[s] hostnames to yield IP addresses.” (Req. at 39-40.) The first and third methods cited by the Request are disclosed as not being performed by the TCP/IP stack, and there is no suggestion that a subsequent connection would be made through a SOCKS server, much less be encrypted. (*See Aventail v3.1* 11, 45.) The second method, highlighted by the Request, is directed to *Aventail Connect* being configured to “send the hostname to a DNS server on another computer for resolution.” (Req. at 40.) It appears that this is the method that the Request contends discloses a DNS request.

33. However, page 12 of *Aventail* describes, in Step 1, that “*Aventail Connect* will forward the hostname to the extranet (SOCKS) server in step 2 and the SOCKS server performs the hostname resolution.” (*Id.* at 12.) In Step 2, after “the connection is completed,” and authentication processing executed, *Aventail Connect* “then sends the proxy request to the extranet (SOCKS) server [including] the DNS entry (hostname) provided in step 1.” (*Id.*) Accordingly, what the Request points to as the request for an IP address is generated by *Aventail Connect* (from the client computer) after the establishment of what the Request points to as the VPN. Consequently, the purported VPN

is not initiated in response to a determination that the DNS request is requesting access to a secure target web site.

34. It is my understanding that the Request also contends that the purported VPN is automatically established when traffic is proxied into a private network as a result of a determination made by Aventail Connect that the traffic should be redirected to the private network. (Req. at 42-43.) However, the cited portion of *Aventail* discloses that Aventail Connect proxies traffic into the private network “[d]epending on the security policy and the Aventail ExtraNet Server configuration.” (*Aventail v3.1* 77.) The Request has not provided any reasoning as to how proxying a connection into a private network based on a policy or configuration includes a determination related to a DNS request.

35. The Request further contends that a VPN is automatically established because the Aventail ExtraNet Server requires “all users to use Aventail Connect to authenticate and encrypt their sessions before any connection to the internal private network(s).” (Req. at 43 (citing *Aventail v3.1* 78).) In my opinion, this just indicates that the encryption of *Aventail* occurs in response to receiving a connection, and does not teach any link between a *DNS request* and the encryption, much less that it is automatically initiated in response to determining that the DNS request in step (2) is requesting access to a secure target web site, as recited by claim 1.

36. For these reasons, it is my opinion that the purported VPN of *Aventail* would not be automatically initiated in response to determining that a DNS request is requesting access to a secure target web site.

37. With regard to claim 4, it is my opinion that *Aventail* does not disclose that step (3) of claim 1 comprises the step of, prior to automatically initiating the VPN between the client computer and the target computer, determining whether the client computer is authorized to establish a VPN with the target computer and, if not so authorized, returning an error from the DNS request. It is my understanding that the Request points to the Fratto declaration (Ex. E2) to try to show that the feature of claim 4 is disclosed by *Aventail* because *Aventail* “would inherently know how to handle errors returned according to the relevant DNS and TCP/IP communication protocols.” (Req. at 47 (citing Ex. E2 (Fratto) at 136-42.) Nothing in *Aventail* discloses returning an error from a DNS request. The declaration discusses how a “DNS response” could be returned with a RCODE when the server refuses to provide a response due to a policy restriction (*i.e.*, the client computer is not authorized). (Req. at 47.) However, as previously explained, *Aventail* teaches that the server does not even attempt to resolve a hostname until after the client computer is authenticated. (*Aventail v3.1* 12 (step

2.b.) Accordingly, the RCODE could not be returned in response to a client computer not being authorized to establish the purported VPN.

38. With regard to claim 6, it is my opinion that *Aventail* does not disclose that step (3) of claim 1 comprises the step of establishing the VPN by creating an IP address hopping scheme between the client computer and the target computer. The Request again points to the Fratto declaration to try to show that the feature of claim 6 is disclosed by *Aventail* because *Aventail* discloses a MultiProxy scheme and a Proxy Chaining scheme. The Request does nothing to show that a VPN is established by creating an IP address hopping scheme. The proxy schemes disclosed by *Aventail* would not be understood by one of ordinary skill in the art as establishing the purported VPN. Rather, the proxy schemes are implemented merely to satisfy the “need to traverse multiple firewalls.” (*Aventail v3.1* 68.) Providing a mechanism for traversing multiple firewalls does not contribute in any meaningful way towards securing data transmitted over a public network, much less establishing a VPN. (*See id.* at 116.)

39. I understand independent claim 10 to recite, among other things, a “DNS proxy server” that “returns the IP address for the requested domain name if it is determined that access to a non-secure web site has been requested.”

40. I understand the Request contends that Aventail Connect, running on the client computer, may be seen as a “DNS proxy server,” and that the Aventail ExtraNet Server is a gateway computer. (Req. at 54.) It appears that the Request contends that Aventail Connect (the alleged proxy server) will return an IP address if a hostname does not match a redirection rule. (*See* Req. at 54.) I disagree.

41. In my opinion, the Request’s view is contradictory to the Request’s own position that “the term ‘DNS proxy server’ means ‘a computer or program that responds to a domain name inquiry in place of a DNS.’” (Req. at 36 (emphasis added).) Aventail Connect does not respond to a domain name inquiry in place of a DNS when a redirection rule is not matched, much less return an IP address for the requested domain name if it is determined that access to a non-secure web site has been requested. *Aventail* teaches that if “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** [which] performs the lookup **as if Aventail Connect were not running.**” (Req. at 54 (quoting *Aventail v3.1* 11) (emphasis in original). *See also* *Aventail v3.1* 45 (hostnames “are passed to the local resolver for name resolution instead of being proxied through the SOCKS v5 server.”).) Thus, *Aventail* discloses that Aventail Connect does not return the IP address for the requested domain

name if it is determined that access to a non-secure web site has been requested. Rather, Aventail Connect is ignored completely.

42. For these reasons, it is my opinion that *Aventail* does not disclose the feature of a DNS proxy server that returns the IP address for the requested domain name if it is determined that access to a non-secure web site has been requested, as recited by claim 10.

43. It is also my opinion that *Aventail* does not disclose a “DNS proxy server that generates a request to create a VPN between the client computer and the secure target computer if it is determined that access to a secure web site has been requested,” as recited by claim 10.

44. The Request contends that, if a hostname matches a redirection rule, “Aventail Connect will . . . begin the TCP handshake with the server . . . .” (Req. at 54 (quoting *Aventail v3.112*)). As an initial matter, the Request mischaracterizes *Aventail*, as the quoted feature is performed only when “the request contains a routable IP address,” and thus the purported proxy server would not receive a request from the client computer to look up an IP address from which a determination may be made. (See *Aventail v3.1 12* (step 2.a).) Even so, Aventail Connect would not be seen as making a request to create a VPN. Aventail Connect merely makes a connection request. (*Aventail v3.1 11-12*.) The connection request of *Aventail* is not understood as a request to create a VPN. Rather, *Aventail* teaches that the step of establishing a connection is no different than in standard Winsock communications, which do not incorporate security. (*Aventail v3.1 8, 11*.) *Aventail* may forward a hostname with the connection request. (*Aventail v3.1 12* (the hostname is forwarded to the extranet server and the server performs the resolution).) *Aventail*, however, does not disclose that the proxy server does anything with the hostname other than perform a hostname resolution. (See *id.*)

45. I understand independent claim 13 to recite “a central computer that maintains a plurality of authentication tables each corresponding to one of the client computers” and “authenticating, with reference to one of the plurality of authentication tables, that the request received in step (1) is from an authorized client.”

46. The Request contends that the Aventail ExtraNet Server inherently maintains authentication tables that are used to authenticate each client computer based on presentation of authentication credentials specific to the client computer. (Req. at 56.) I disagree that *Aventail* discloses that this feature is shown to be present. Authentication tables are not required to authenticate a client computer. Alternatives to authentication tables are known in the art for authenticating a client computer.

47. Indeed, several mechanisms related to authentication are disclosed by *Aventail*, including local and remote configuration files, (*id.* at 30), client authentication modules, (*id.* at 46-48), credentials stored in memory or on a disk, (*id.* at 50), or an exchange of certificates between clients and servers, (*id.* at 52-54, 59), none of which are authentication tables. Alternatively, a client may be authenticated using a security token, in which a client has a pseudo-random number generator that is synchronized with a pseudo-random number generator at an authenticator. In this method, the client sends a security token (pseudo-random number) generated by its pseudo-random number generator to the authenticator. The authenticator authenticates the client by comparing the security token from the client with a pseudo-random number generated by its pseudo-random number generator at the time of receipt of the security token. This authentication method does not require that the authenticator maintain an authentication table.

48. With regard to claim 14, it is my opinion that *Aventail* does not disclose at least the feature of communicating according to a scheme by which at least one field in a series of data packets is periodically changed according to a known sequence. *Aventail* does not disclose changing a field in a series of data packets, much less communicating according to a scheme by which at least one field in a series of data packets is periodically changed according to a known sequence. *Aventail* also does not disclose how IP header information would be altered in a MultiProxy scheme or a Proxy Chaining Scheme, as suggested by the Request, or how that alteration would disclose the specific feature of claim 14, *e.g.*, how one field in a series of data packets is periodically changed according to a known sequence. *Aventail* does not even disclose IP header information.

**B. Wang**

49. *Wang* discloses four connection architectures: (1) Transparent ATM Core Network architecture; (2) L2TP Access Aggregation (LAA) architecture; (3) PPP Terminated Aggregation (PTA) architecture; and (4) Virtual Path Tunneling Architecture (VPTA). (*Wang* 13) In the Transparent ATM Core Network architecture, ATM layer connectivity is established between the customer premises and the Network Service Provider (NSP). (*Id.*) The session setup and release phases at the link level and network level are specified by the NSP. (*Id.*)

50. In the LAA architecture, a PPP session of the user is extended to a remote network server over an arbitrary network. (*Id.* at 13.) To this end, the LAA architecture includes a L2TP Access Concentrator (LAC) and a L2TP Network Server (LNS). (*Id.* at 14.)

51. During session establishment, the user utilizes a dialer application on the Customer Premises Equipment (CPE) to initiate a session by establishing a PPP (point-to-point protocol)

connection between the user's CPE and the LAC. (*Id.*) The PPP allows for authentication to be requested during PPP negotiation. (*Id.* at 15.) For this to work, the LAC needs to be informed of the user's intended NSP. (*Id.*) To this end, the user's CPE sends a user name along with a fully qualified domain name to the LAC during an PPP authentication phase. (*Id.*)

52. Based on the user name and domain information provided in the PPP authentication phase, the LAC determines the destination NSP. (*Id.*) For example, if the user name is "Joe@nsp.net," the LAC knows that "nsp.net" is the destination NSP. (*Id.*) Once the destination NSP is identified, the LAC determines if a tunnel already exists to the proper LNS. If it does not exist, the LAC establishes one. (*Id.*)

53. In the PTA architecture, the PPP sessions are not tunneled all the way to the NSP, as in the LAA architecture. (*Id.* at 16.) Instead, the PPP sessions are terminated in a Broadband Access Server (BAS). (*Id.*) At the BAS, the packets are extracted and forwarded over a Regional Broadband Network to the proper NSP. (*Id.*)

54. The user utilizes a dialer application on the CPE to initiate a session by establishing a PPP connection between the user's CPE and the BAS. (*Id.* at 18.) The dialer program is configured to facilitate PPP negotiation and login ID and password entry. (*Id.*) Upon establishment of the session with the BAS, the BAS initiates the authentication stage and challenges the user for the user name and password. (*Id.*) As with the LAA model, the user will reply back with a user name along with a fully qualified domain name. (*Id.*) The BAS extracts the domain string portion of the user name and sends a query to the NSP to authenticate and obtain address information (*e.g.*, DNS server's address). (*Id.*) In the case of IP network, the NSP replies with an IP address and other IP configuration information (*e.g.*, DNS server's IP address). (*Id.*)

55. The VPTA is similar to the Transparent ATM architecture, except that the PPP session uses a SVC established between CPE and the Access Node or a proxy signaling device. (*Wang* 19.)

56. With regard to claim 1, the Request asserts that *Wang* discloses "(1) generating from the client computer a Domain Name Service (DNS) request that requests an IP address corresponding to a domain name associated with the target computer." In support of its assertion, the Request cites several passages of *Wang* describing the LAA, PTA and VPTA architectures of *Wang*. (Req. at 121, 122.) I disagree. In my opinion, none of the passages cited by the Request discloses "an IP address corresponding to a domain name associated with the target computer," let alone a request for such an address.



57. The cited passages of *Wang* relating to the LAA architecture do not mention an IP address.

58. The cited passages of *Wang* relating to the PTA architecture disclose the NSP replying to a query from the BAS “with an IP address and other IP configuration information (*e.g.*, DNS server’s IP address).” (Req. at 122 quoting *Wang* 18.) Thus, the cited portions of *Wang* relating to the PTA architecture disclose a “DNS server’s IP address,” not “an IP address corresponding to a domain name associated with the target computer,” as recited in claim 1.

59. I understand that the Request further asserts that *Wang* discloses “(2) determining whether the DNS request transmitted in step (1) is requesting access to a secure web site” in the LAA and PTA architectures of *Wang*. (Req. at 123 and 124). The Request also asserts that the domain name provided to the LAC or BAS of *Wang* by the user constitutes the DNS request recited in claim 1. (*Id.*) I disagree.

60. In the LAA architecture of *Wang*, “a user name along with a fully qualified domain name” is provided to the LAC during a PPP authentication phase. (*Wang* 15.) “Based on the user-name and domain information provided in the authentication phase of the PPP establishment, the LAC determines the destination.” (*Id.*) “For example, if the user enters Joe@nsp.net for the user-name, the LAC knows that *nsp.net* is the destination NSP.” (*Id.*) The LAC then “determines if a tunnel already exists to the proper LNS.” (*Id.*) “If it does not exist, the LAC establishes one.” (*Id.*)

61. Nowhere in the description of the LAC does *Wang* teach or suggest the LAC performing the step of “determining whether the DNS request transmitted in step (1) is requesting access to a secure web site,” as recited in claim 1.

62. In the PTA architecture of *Wang*, the user provides the BAS “with a user-name along with a fully qualified domain name.” (*Wang* 18.) “The BAS extracts the domain string portion of the user-name and sends off a query to NSP to authenticate and obtain address information (*e.g.*, DNS server’s address).” (*Id.*)

63. Likewise, nowhere in the description of the BAS does *Wang* teach or suggest the BAS performing the step of “determining whether the DNS request transmitted in step (1) is requesting access to a secure web site,” as recited in claim 1.

64. In support of its assertion that *Wang* discloses the above features of claim 1, the Request contends that in, the LAC and BAS examples, “a domain name supplied with the user credentials is evaluated to determine if a request is being made to establish a connect to a secure destination (*e.g.*, the corporate network).” (Req. at 124.) I disagree.

65. *Wang* discloses that the LAC or BAS evaluates the domain name to determine the destination NSP so that the LAC can create a tunnel to the proper LNS or the BAS can send a query to the destination NSP. In my opinion, this does not require that the LAC or BAS determine whether the destination is a secure destination.

66. The Request contends that the term “virtual private network” means “a network of computers which privately communicate with each other by encrypting traffic on insecure communication paths between the computers.” (Req. at 35 (citing *VirnetX v. Microsoft*, 6:07cv80 (EDTX).) Thus, the Request’s definition of VPN requires, at a minimum, encryption of traffic.

67. The Request also contends that the creation of a tunnel by the LAC of *Wang* discloses automatically initiating a VPN between the client computer and a target computer. (Req. at 124.) In view of the Request’s own definition of a VPN, the creation of a tunnel by the LAC alone does not constitute initiation of a VPN. A tunnel by itself does not provide encryption. A tunnel passes traffic therethrough regardless of whether the traffic is encrypted or not.

68. Further, the Request’s alleged automatic initiation of a VPN (i.e., creation of a tunnel by the LAC) is inapplicable to the PTA architecture of *Wang* since the BAS does not tunnel to the NSP. This is clear from *Wang*’s disclosure that, in the case of the BAS, “instead of being tunneled all the way to the NSP, the PPP sessions are terminated in the Broadband Access Server (BAS).” (*Wang* 16.)

69. The Request asserts that the features recited in claim 4 are disclosed in section 9.2 on page 20 of *Wang*, titled “Authentication, Authorization and Accounting.” (Req. at 129.) I disagree.

70. Section 9.2 of *Wang* discloses the LAC forwarding authentication information to the NSP to perform authentication. (*Wang* 20). However, in the rejection of claim 1, the Request asserts that the automatic creation of a tunnel by the LAC is “automatically initiating the VPN.” (Req. at 124.) Thus, the Request’s alleged determination whether the user is authorized to establish a VPN (authentication by the NSP) occurs after, not prior to, the Request’s alleged automatic initiation of the VPN (creation of the tunnel by the LAC).

71. In my opinion, *Wang* is silent regarding “returning an error from the DNS request,” as recited in claim 4.

72. With regard to claim 5, the Request contends that “as shown on page 15 [of *Wang*] a user will have to have authenticated successfully during “the authentication phase of the PPP establishment” before the domain information is sent to the LAC to be resolved (i.e., before “the LAC determines the destination”).” (Req. at 130.) In my opinion, the Request’s contention about

the disclosure of *Wang* is incorrect.

73. *Wang* discloses that “a user name along with a fully qualified domain name [is] entered during the PPP authentication phase.” (*Wang* 15.) This is inconsistent with the Request’s contention that a user will have to have authenticated successfully during the PPP authentication phase before the domain name is sent to the LAC.

74. In my opinion, *Wang* also does not disclose “determining whether the client computer is authorized to resolve addresses of non secure target computers” during the PPP authentication phase. Thus, even if the Request’s contention were correct, *Wang* still would not meet claim 5.

75. In my opinion, *Wang* is silent regarding “returning an error from the DNS request,” as recited in claim 5.

76. With regard to claim 6, the Request contends the feature “wherein step (3) comprises the step of establishing the VPN by creating an IP address hopping scheme between the client computer and the target computer” recited in claim 6 is disclosed on pages 16 and 21 of *Wang*. (Req. at 130, 131.) I disagree.

77. The cited portions of *Wang* do not disclose hopping between different IP addresses. In my opinion, this means that *Wang* cannot disclose “establishing the VPN by creating an IP hopping scheme between the client computer and the target computer,” as recited in claim 6.

78. With regard to claim 10, the Request asserts that *Wang* discloses “a DNS proxy server that receives a request from the client computer to look up an IP address for a domain name, wherein the DNS proxy server returns the IP address for the requested domain name if it is determined that access to a non-secure web site has been requested,” as recited in claim 10. More particularly, the Request asserts that the LAC or BAS of *Wang* discloses the DNS proxy server recited in claim 10, and suggests that the domain name provided to the LAC or BAS by the user constitutes a DNS request. (Req. at 136, 137.)

79. With regard to the LAC, *Wang* does not disclose the LAC returning an IP address. In my opinion, this alone precludes the LAC from disclosing the DNS proxy server recited in claim 10.

80. With regard to the BAS, *Wang* discloses the NSP replying to a query from the BAS “with an IP address and other IP configuration information (e.g., DNS server’s IP address).” (*Wang* 18.) Therefore, *Wang* discloses the BAS receiving a “DNS server’s IP address” from the NSP, and not the IP address for the domain name provided by the user. In my opinion, this alone precludes the BAS from disclosing the DNS proxy server recited in claim 10.

81. In my opinion, *Wang* does not disclose that the LAC or the BAS of *Wang* “generates

a request to create the VPN between the client computer and the secure computer” for the following reasons.

82. Referring to the LAC and BAS examples of *Wang*, the Request contends “*Wang* in these examples show VPNs (*i.e.*, secure encrypted **tunnels** being established after authentication) being established in response to DNS request that specify a secure domain (*e.g.*, a domain name corresponding to the corporate network).” (Req. at 137 (emphasis added).) I disagree.

83. In my opinion, the Request’s assertion equates VPNs with “secure encrypted tunnels.” But the Request’s VPNs (“secure encrypted tunnels”) are inapplicable to the BAS of *Wang*. This is because the BAS does not tunnel to the NSP. This is clear from *Wang*’s disclosure that, in the case of the BAS, “instead of being tunneled all the way to the NSP, the PPP sessions are terminated in a Broadband Access Server (BAS).” (*Wang* 16.)

84. With regard to the LAC, *Wang* discloses that the LAC creates a tunnel to the proper LNS if one does not already exist. The creation of the tunnel alone does not constitute creation of the Request’s alleged VPN (“secured encrypted tunnel”). This is because a tunnel by itself does not provide encryption, but merely passes traffic therethrough regardless of whether the traffic is encrypted or not.

85. With regard to claim 13, authentication tables are not required to authenticate a client computer. Alternatives to authentication tables are known in the art for authenticating a client computer.

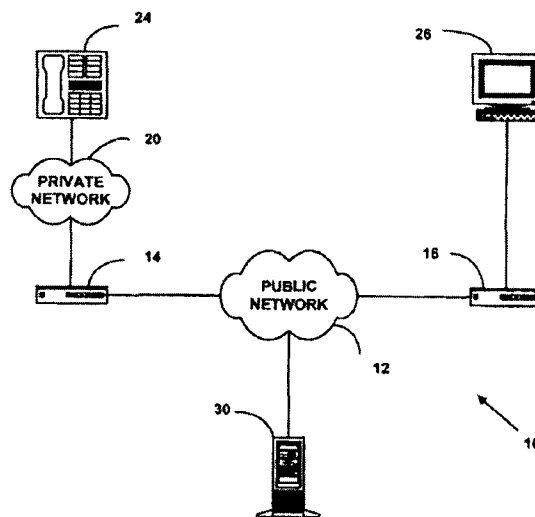
86. For example, a client may be authenticated using a certificate authority or Lightweight Directory Access Protocol (LDAP), an application protocol for accessing and maintaining distributed directory information services over a network. Directory information may be stored in a directory information tree (DIT) and have no resemblance to a table structure, much less an authentication table.

87. In another example, a client may be authenticated using a security token, in which a client has a pseudo-random number generator that is synchronized with a pseudo-random number generator at an authenticator. In this method, the client sends a security token (pseudo-random number) generated by its pseudo-random number generator to the authenticator. The authenticator authenticates the client by comparing the security token from the client with a pseudo-random number generated by its pseudo-random number generator at the time of receipt of the security token. This authentication method does not require that the authenticator maintain an authentication table.

C. *Beser* in view of *Kent*

88. *Beser* discloses a system for initiating a tunneling connection that hides the identity of the originating and terminating ends of the tunneling association from other users. (*Beser* Abstract.) With reference to Fig. 1, reproduced below, *Beser* discloses that a first network device 14 informs a trusted-third-party network device 30 of a request to initiate a tunneling connection received from an originating telephony device 24. (*Beser* 7:62-8:4, 10:2-6, 11:9-10.)

FIG. 1



89. The request to initiate a tunneling connection includes a unique identifier for a terminating telephony device 26. (*Id.* at 10:4-6.) After being informed of the request, trusted-third-party network device 30 associates an identifier of terminating telephony device 26 with a public IP address of a second network device 16. (*Id.* at 11:26-32.) Then, private IP addresses for each of the originating telephony device 24 and the terminating telephony device 26 are negotiated and distributed to the second network device 16 and the first network device 14, respectively. (*See, e.g., id.* at 11:59-12:54.) This way, the tunneling connection “hides the identity of the originating and terminating ends of the tunneling association from the other users of the public network.” (*Id.* at 2:36-39.)

90. *Beser* is directed towards “initiating a tunneling association” and in some aspects, a “virtual tunnel,” primarily in the context of voice-over-IP (“VoIP”) communications. (*Beser* 6:58-59.) *Beser* does not disclose encrypting traffic on insecure communication paths via tunneling. To the contrary, *Beser* explains that encryption is “infeasible” and/or “inappropriate” in VoIP applications. (*Beser* 1:58-2:17.)

91. *Kent* is a Request for Comments (RFC) that discloses IPsec, a type of security

protocol. (Req. at 163.) The Request contends that *Beser* could be modified with the security protocol disclosed by *Kent* to form the virtual private network of claim 1. (*Id.* at 164.) The Request contends that *Beser* “shows that the IPsec protocol was known as being useful for encrypting traffic in IP tunnels,” and that “the IPsec protocol can and should be used to handle the encryption of the traffic being sent through the IP tunnel.” (*Id.* at 163.)

92. *Beser*, however, discloses IPsec only to the extent that it has been used to protect the information inside IP packets. To the contrary, *Beser* specifically teaches against using IPsec and other encryption techniques in tunneled connections and VoIP applications, the technology of which *Beser* is primarily concerned. (*Beser* 1:54-67.) *Beser* takes the position that such encryption may be “infeasible” for VoIP due to system strain on computing power and increased investment in VoIP equipment, and “inappropriate” for the transmission of multimedia or VoIP packets. (*Beser* 1:58-2:17.) For these reasons, it is my opinion that *Beser* discloses a system and method directed to initiating a tunneling association, in which IP packets are not encrypted because of the problems with encryption. (*See id.* at 2:36-40.)

93. *Kent* discloses IPsec, the very protocol that *Beser* explicitly teaches as being problematic. *Beser*’s disclosed system and method for initiating a tunneling association is intended as an alternative to encryption to address the drawbacks that arise from the teachings of *Kent* (e.g., high computing power), not to encourage the use of encryption. Accordingly, one of ordinary skill in the art would not have combined *Beser* with *Kent* in order to achieve the purported VPN (including encryption).

94. Additionally, for the following reasons, it is my opinion that one skilled in the art, on reviewing *Beser* and *Kent*, would not contemplate the feature of initiating a VPN in response to determining that a DNS request is requesting access to a secure target web site, as recited by claim 1.

95. I understand that the Request asserts that *Beser* teaches in response to a request containing a unique identifier specifying the location of a second device, a trusted-third-party device will negotiate with first and second network devices to establish an IP tunnel between the first and second devices. (Req. at 165.) *Beser* does not disclose that encryption would be initiated in response to the unique identifier described by *Beser*. *Beser*, in step 114, merely discloses that the unique identifier may be encrypted before the identifier is sent to the trusted-third-party device, to then be associated with an address of a second network device 16 at step 116. (*See Beser* 11:22-28 (“IP 58 packets may require encryption or authentication to ensure that the unique identifier cannot be read on the public network.”) (Emphasis added).) *Beser* does not show computers that privately

communicate with each other by encrypting traffic on insecure communication paths, much less initiating the purported VPN, in response to determining that the DNS request is requesting access to a secure target web site.

96. Additionally, *Beser* discloses that “[o]ther possibilities are that the unique identifier ... is a domain name and may be used to initiate the VoIP association.” (*Id.* at 10:55-57. *See also* Req. at 153 (citing *Beser*.) Accordingly, if the unique identifier (*e.g.*, a domain name) initiates the VoIP association then the unique identifier would be sufficient by itself, without further action, to initiate the VoIP association. Consequently, in my opinion, the unique identifier is not shown to be a DNS request that requests an IP address. In this configuration, the system of *Beser* simply receives the unique identifier and initiates the VoIP association. Because there is no DNS request that requests an IP address, encryption cannot be initiated in response to a determination related to a DNS request. Moreover, the Request has not shown an example of how a determination may be made if the VoIP association is initiated by the unique identifier.

97. In my opinion, *Beser* discloses a request to initiate a VoIP association, not a DNS request that requests an IP address corresponding to a domain name associated with the target computer, as recited by claim 1. (*Beser* 10:2-3.) *Beser* discloses that the request may include a unique identifier, (*id.* at 8:1-3, 10:5-6), which may be, in some instances, a domain name. (*Id.* at 10:41.) However, in my opinion, merely including a domain name in the request to initiate a VoIP association does not transform it into a request for an IP address.

98. It is my understanding that claim 1 also recites determining whether the DNS request transmitted in step (1) is requesting access to a secure web site. The Request asserts that *Beser* discloses “a trusted-third-party device will receive and then evaluate a request, compare it to a database of entries, and take additional actions to establish the IP tunnel based on the results of that evaluation.” (Req. at 164.) The Request then asserts that *Beser* shows that a determination is made whether a domain name is specifying a secure destination because the trusted-third-party-device inherently will not route a request to an unknown destination. (*Id.*) I disagree.

99. First, *Beser* merely discloses that the trusted-third-party device may retain a list of E.164 numbers of its subscribers. (*Beser* 11:45-50.) A list of numbers does not characterize any of the numbers, and certainly does not disclose that one or more of those numbers may be for a secure destination and others are not. *Beser* simply does not disclose that this list of numbers has any purpose related to security.

100. Second, any “determination” in *Beser* would be based on whether a request was for

an “unknown” destination. In view of this, if the destination is “unknown” then there would be no way to characterize the destination, and certainly not to determine that the request was for a secure destination.

101. With regard to claim 3, the Request asserts that the trusted-third-party network device of *Beser* functions as a DNS server, and will inherently resolve and return an IP address. (Req. at 167.) *Beser*, however, does not disclose that trusted-third-party network device will return an IP address to a client computer if it is determined that access to a non secure website is being requested. In the proposed rejection of claim 1, the Request suggests that a non-secure website is one that is “unknown to the third-party-network device.” (*Id.* at 164.) If this is the case, then the trusted-third-party network device could not return an IP address of the purported non-secure website because (by virtue of the website being unknown) the trusted-third-party network device would not have an IP address to return. Besides, *Beser* does not disclose what would happen if the requested website is unknown (*i.e.*, not “secure” according to the Request).

102. With regard to claim 4, I understand the Request asserts that *Beser* discloses authorization, and a failure of authentication will result in no establishment of the IP tunnel, and therefore discloses returning an error to a DNS request. (*See* Req. at 167.) I disagree. *Beser* does not disclose determining whether a client computer is authorized to establish a VPN. *Beser* merely discloses that an IP packet may be encrypted or authenticated “to ensure that the public IP 58 address of the second network device 16 cannot be read on the public network.” (*E.g.*, *Beser* 11:22-25.) Encrypting or authenticating an IP packet to prevent non-authenticated computers from viewing an IP address does not disclose determining whether a client computer is authorized to establish a VPN with the target computer. *Beser* also does not disclose returning an error in response to a request to initiate a VoIP connection, and certainly not in response to a DNS request. In my opinion, even if *Beser* could be shown to not establish an IP tunnel if a certain device is not found, *Beser* does not disclose that an error would be returned. The Request does not explain how *Kent* is relevant to claim 4. Even so, *Kent* is merely directed to IPsec, and is not concerned with DNS requests. *Kent* merely discloses ICMP error messages. (*See, e.g.*, *Kent* 37 § 5.2.2.) ICMP error messages are generally related to providing a message when a service is not available or a host or router could not be reached, and are not concerned with DNS requests.

103. With regard to claim 8, the Request points to a brief disclosure in *Beser* that a third-party-network device can be a domain name server, and contends that the trusted-third-party network device is functioning as a DNS server. (*Id.* at 169 (citing *Beser* 11:32-36).) The Request also



contends that the trusted-third-party network device will, by its nature of being a DNS server, simply return the IP address associated with a non-secure destination. (*Id.* at 169.) However, nothing in the Request describes how the trusted-third-party network device could function as a DNS proxy server that passes through the request to a DNS server. Indeed, *Beser* does not disclose that this server may function as a DNS proxy server that passes through the request to a DNS server. *Beser* merely discloses that the third-party-network device may be a domain name server.

104. With regard to claim 10, The Request asserts that *Beser* discloses a DNS proxy server because (1) *Beser* discloses a trusted-third-party network device that can be a domain name server; (2) “[d]omain name servers were known to inherently function by evaluating domain names, and returning IP addresses associated with the domain,” and (3) the third-party-network device of *Beser* “will receive and then evaluate a request . . .” (Req. at 171.) I disagree for similar reasons as those I provided for claim 8. For reasons similar to claim 8, the *Beser* does not disclose a DNS proxy server, a DNS proxy server that receives a request from the client computer to look up an IP address for a domain name, determining whether a web site is secure, returning an IP address for a requested domain name if it is determined that access to a non-secure web site has been requested, and creating a VPN between the client computer and the secure target computer if it is determined that access to a secure web site has been requested.

105. With regard to claim 12, it is my opinion that *Beser* does not disclose that a gatekeeper computer determines whether the client computer has sufficient security privileges to create the VPN and, if the client computer lacks sufficient security privileges, rejecting the request to create the VPN. *Beser* merely discloses that an IP packet may be encrypted or authenticated “to ensure that the public IP 58 address of the second network device 16 cannot be read on the public network.” (*E.g.*, *Beser* 11:22-25.) Encrypting or authenticating an IP packet to prevent non-authenticated computers from viewing an IP address does not disclose determining whether a client computer has sufficient security privileges to create a VPN.

106. The Office Action, or the Request, does not specify how claim 13 may be contemplated by *Beser* in view of *Kent*. The Request includes claim 13 in a heading above a brief summary argument as to why a substantial new question or patentability has been raised. (Req. at 29.) However, that summary argument is directed only toward how *Beser* and *Kent* may be combined to show the subject matter (*i.e.*, a VPN) of claims 1 and 10. The argument does not describe how *Beser* or *Kent* teaches at least the feature of a plurality of client computers, authenticating, with reference to one of the plurality of authentication tables, or allocating resources

to establish a virtual private link between the client and a second computer, as recited by claim 13.

107. I have reviewed the rejection of claim 18. Claim 18 is missing from the Request's detailed explanation of how *Beser* and *Kent* may be applied in any rejection. (*See, e.g.*, Req. at 173-74.) The summary arguments do not describe how *Beser* or *Kent* teaches the features of claim 18, including determining whether the client computer is authorized to resolve addresses of non secure target computers and, if not so authorized, returning an error from the DNS request. Therefore, I am unable to understand why claim 18 has been rejected. Nevertheless, it is my opinion that one would not contemplate the features of claim 18 for at least the same or similar reasons previously given with respect to claim 1 and claim 4.

**D. *Beser* in view of *Kent*, in further view of *Blum***

108. Regarding claim 3, it is my opinion that one of ordinary skill in the art, on reading *Blum* in combination with *Beser* or *Kent*, would not contemplate the step of: (4) in response to determining that the DNS request in step (2) is not requesting access to a secure target web site, resolving the IP address for the domain name and returning the IP address to the client computer. *Blum* is directed to a method and apparatus for providing transparent proxy services. (*E.g.*, *Blum* 2:25-26.) A "proxy server," as defined by *Blum*, is an application that provides access to the Internet or other external network, and evaluates requests and determines which of the requests to pass on to the Internet or other network. (*Id.* at 1:10-25.) *Blum* enables communications from a client computer, through a proxy, to remote DNS. (*Id.* at 6:40-43.) However, *Blum* does not disclose returning an IP address or determining whether a DNS request is requesting access to a secure target web site, and thus does not add anything to the domain name server disclosed by *Beser*. (*See Beser* 11:33-34.)

109. Regarding claim 5, it is my opinion that one of ordinary skill in the art, on reading *Blum*, in combination with *Beser* or *Kent*, would not contemplate the feature of step (3) comprises the step of, prior to automatically initiating the VPN between the client computer and the target computer, determining whether the client computer is authorized to resolve addresses of non secure target computers and, if not so authorized, returning an error from the DNS request. As previously noted, the Request suggests that a non-secure website is one that is "unknown to the third-party-network device." (*Id.* at 164.) If this is the case, the unknown website could not be resolved because it is "unknown." For this reason alone, in my opinion, *Beser* is incompatible with any reference that would otherwise disclose resolving an address of a "non-secure" website as allegedly taught by *Beser*. One of ordinary skill would not have looked to vary the disclosure of *Beser* by adding the

ability to determine whether the client computer is authorized to resolve addresses of unknown websites, much less non-secure target computers. Besides, as previously explained, *Beser* does not disclose returning an error in response to a request to initiate a VoIP connection, much less in response to a DNS request. Even if *Beser* could be shown to not establish an IP tunnel if a certain device is not found, *Beser* does not disclose that an error would be returned.

110. The Request also does not explain how *Kent* is relevant to claim 5. *Kent* is merely directed to IPsec, and is not concerned with DNS requests. *Kent* just discloses ICMP error messages. (See, e.g., *Kent* 37 § 5.2.2.) ICMP error messages are generally related to providing a message when a service is not available or a host or router could not be reached, and are not concerned with DNS requests. Similar to *Kent*, *Blum* discloses returning an error message if a DNS request fails because DNS services available to the client computer were not able to resolve the request, not because the client computer was not authorized. (*Blum* 8:65-9:3.) Therefore, *Blum* does nothing to show that an error could be returned on a failure of authentication, much less when a client computer is not authorized to resolve addresses of non secure target computers. (*Beser* 11:33-34.)

111. With regard to claim 8, *Blum* discloses a proxy that enables communications from a client computer, through a proxy, to a remote DNS. (*Blum* 6:40-43.) However, *Blum* does not disclose enabling those communications based on a determination of whether a DNS request is requesting access to a secure website. Indeed, the Request merely asserts that *Blum* describes a proxy server that determines if DNS requests require a remote connection. (Req. at 175 (citing *Blum* 6:40-57).) Determining whether a remote connection is required does not disclose determining whether a DNS request is requesting access to a secure web site.

112. For the same or similar reasons given above, it is my opinion that one of ordinary skill in the art, on reading *Beser*, *Kent*, and *AutoSOCKS*, would also not contemplate the features of claims 3, 5, 8, and 9.

**E. *BinGO***

113. *BinGO* is directed to a router (*BinGO!* router) used to connect a user with the network of an Internet provider or to a company's head office from the user's home or branch office via integrated services digital network (ISDN). (See *BinGO UG* 15-16, Figure 1-1.) *BinGO* states that if the user wants to access the Internet, the user must set up the user's Internet service provider (ISP) as a wide area network (WAN) partner on the *BinGO!* router, and if the user wishes to establish a local area network (LAN) to LAN connection (e.g., between the user's LAN and the LAN of the user's head office), the user must configure the LAN of the user's head office as a WAN partner. (*Id.* at

143.) *BinGO* further describes how to configure the *BinGO!* router (e.g., by using a configuration wizard) such that it may be connected to a WAN partner such as a corporate network. (*Id.* at 45, 53-54.)

114. In configuring the *BinGO!* router, encryption for a connection to the WAN partner may be selected. (*Id.* at 149-150, 175.) While *Bingo UG* describes that the connection to the WAN partner (e.g., either the ISP or the corporate network) may be encrypted, *BinGO EFR* describes alleged VPNs that are established only through an ISP. In particular, *BinGO EFR* describes two scenarios for establishing an alleged VPN: a PPTP Client-to-VPN Server scenario and a LAN-to-LAN VPN scenario. (See *BinGO EFR* 83-84.) Under either scenario, however, *BinGO EFR* describes that a connection is established to the local ISP first, and then an alleged VPN is established over the Internet. (*Id.* at 83-84).

115. I understand the Request contends: 1) that a user's PC connected to the *BinGO!* router on a LAN corresponds to the client computer recited in independent claim 1; 2) that entering the computer name, "BossPC," at the user's PC corresponds to the DNS request recited in independent claim 1; 3) that the corporate network corresponds to the secure web site recited in independent claim 1; and 4) that the actual BossPC itself corresponds to the target computer recited in independent claim 1. (See Req. at 190-98.) I also understand the Request also contends that *BinGO* describes three different DNS handling procedures, each of which allegedly corresponds to the feature of determining whether the DNS request transmitted in step (1) is requesting access to a secure web site, as recited in independent claim 1. (*Id.* at 194-98.) It is my opinion that these contentions are erroneous.

116. Even assuming that the user's PC corresponds to the client computer recited in independent claim 1 and that the BossPC corresponds to the target computer recited in independent claim 1, it is my opinion that *BinGO* nevertheless fails to disclose the features of independent claim 1 under each of the three different DNS handling procedures.

117. Under the first alleged DNS handling procedure of *BinGO*, I understand the Request assumes that there is a DNS server on the LAN with the *BinGO!* router that could be used to resolve IP addresses of secure destination computers (e.g., on the corporate network such as the BossPC). (*Id.* at 194.) I also understand the Request assumes that the *BinGO!* router could function as a DNS proxy server, that the *BinGO!* router could be configured to use a primary and secondary DNS server, and that the process of DNS resolution is hierarchical (meaning that the first DNS server would be consulted, and if no IP address was returned, the DNS request would be sent to the

secondary DNS server, etc.). (*Id.* at 194-95.)

118. I understand the Request contends that under the first alleged DNS handling procedure, after receiving the alleged DNS request (*e.g.*, entering the computer name, “BossPC,”) from a client computer (*e.g.*, the user’s PC), the *BinGO!* router would use the local DNS server containing the entries of secure destinations to try to resolve the DNS request. (*Id.* at 195.) I also understand the Request contends that if the DNS query returned an IP address, the *BinGO!* router would automatically establish a connection with the secure destination in the corporate network using the WAN partner configuration settings for that WAN partner (*e.g.*, the corporate network at which the BossPC is located). (*Id.*) Furthermore, I understand the Request contends that if this DNS server did not resolve the address (*i.e.*, because the request did not specify a secure destination), the *BinGO!* router would send the request to a secondary DNS server (*e.g.*, one associated with an ISP designated to be the “default route” in the *BinGO!* router configuration settings). (*Id.*) As discussed below, it is my opinion that *BinGO* fails to disclose the feature of a VPN between the client computer and the target computer recited in independent claim 1.

119. I understand the Request alleges that the VPN between the client computer and the target computer recited in independent claim 1 is disclosed in *BinGO* by referring to the descriptions of alleged VPNs in *BinGO* or to the encrypted connection between the *BinGO!* router and the corporate network. (*Id.* at 186-189.) Nevertheless, if the Request considers the encrypted connection between the *BinGO!* router and the corporate network as corresponding to the VPN between the client computer and the target computer recited in independent claim 1, it is my opinion that the Request has not shown how such an encrypted connection can be considered a VPN, let alone a VPN between the user’s PC (the alleged client computer) and the BossPC (the alleged target computer). The Request simply notes that the *BinGO!* router supports encryption while at the same time generally referring to the descriptions of alleged VPNs in *BinGO*. (See Req. at 186-89.) Then later – when referring to the actual feature of the VPN between the client computer and the target computer recited in independent claim 1 – the Request merely states, “As explained above, *BinGO* describes processes in which VPNs are automatically established by *BinGO!* routers,” without explaining if and/or how the encrypted connection between the *BinGO!* router and the corporate network can be considered a VPN, let alone a VPN between the user’s PC and the BossPC. (*Id.* at 190).

120. Furthermore, if the Request considers the descriptions of alleged VPNs in *BinGO* as corresponding to the VPN between the client computer and the target computer recited in independent claim 1, I understand the Request points to descriptions of alleged VPNs in both *BinGO*

*UG* and *BinGO EFR* as allegedly corresponding to the VPN between the client computer and the target computer recited in independent claim 1. (See Req. at 186-191.) However, neither the descriptions of alleged VPNs in *BinGO UG* nor the descriptions of alleged VPNs in *BinGO EFR* can correspond to the VPN between the client computer and the target computer recited in independent claim 1, as I explain below.

121. Regarding the descriptions of alleged VPNs in *BinGO UG*, *BinGO UG* describes that the *BinGO!* router can set up an alleged VPN:

*BinGO!* can set up a VPN (Virtual Private Network) using the PPTP (Point to Point Tunneling Protocol). This provides safe (encrypted) transmission of data over WAN connections, e.g., over Internet. It can be used, for example, by field service staff to obtain low-cost access to data in the company network via Internet and laptop (dial-in via a local Internet Service Provider.)

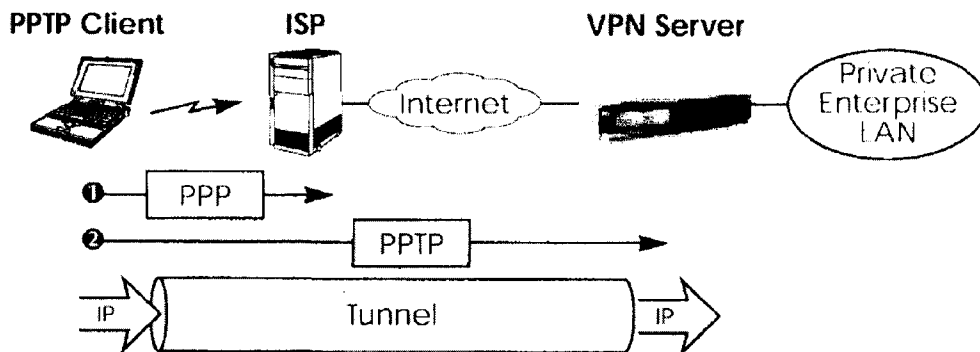
(*BinGO UG* 266.) I understand the Request contends that this description corresponds to the VPN between the client computer and the target computer recited in independent claim 1. (See Req. at 186-87 and 190-91.) I also understand the Request considers the user's PC as corresponding to the client computer recited in independent claim 1, and the BossPC as corresponding to the target computer recited in independent claim 1. (*Id.* at 190-92.) However, in my opinion, nowhere does *BinGO UG* disclose that the alleged VPN that can be set up by the *BinGO!* router is applicable for a connection between the user's PC and the BossPC, which is in the corporate network. *BinGO* only describes that the alleged VPN that can be set up by the *BinGO!* router is over the Internet and via dialing an Internet Service Provider (ISP). However, *BinGO* describes that the *BinGO!* router either connects directly to the Internet (via the ISP) or to the corporate network (without connecting to the ISP). (See *BinGO UG* 15-16, Figure 1-1.) Thus, the alleged VPN that can be set up by the *BinGO!* router is only described to connect to or over the Internet, and not to the corporate network at which the BossPC is located. Therefore, it is my opinion that the descriptions of alleged VPNs in *BinGO UG* cannot correspond to the VPN between the client computer and the target computer recited in independent claim 1.

122. Regarding the descriptions of alleged VPNs in *BinGO EFR*, I understand the Request points to descriptions of alleged VPNs in *BinGO EFR* on pages 76-77, 82, and 84-85 as corresponding to the VPN between the client computer and the target computer recited in independent claim 1. (See Req. at 187-91.) However, nowhere does *BinGO EFR* describe that the virtual private networking described therein is applicable to the *BinGO!* router. *BinGO EFR* states that "[d]epending on your particular product some of the features described in this document may not

be available on your system.” (See *BinGO EFR* 2.) Indeed, nowhere in the entire chapter of *BinGO EFR* describing virtual private networking (e.g., pages 73-98) does *BinGO EFR* mention that the described alleged VPN services are available for the *BinGO!* router. *BinGO EFR* just describes that the alleged VPN services are available for a “BRICK” device and a “BRICK VPN Server.” (See, e.g., *BinGO EFR* 74-78, 80, 81, 84, 85, 87-98.) Therefore, it is my opinion that the alleged VPN services described in *BinGO EFR* are only available for the BRICK router, not the *BinGO!* router described in *BinGO UG*.

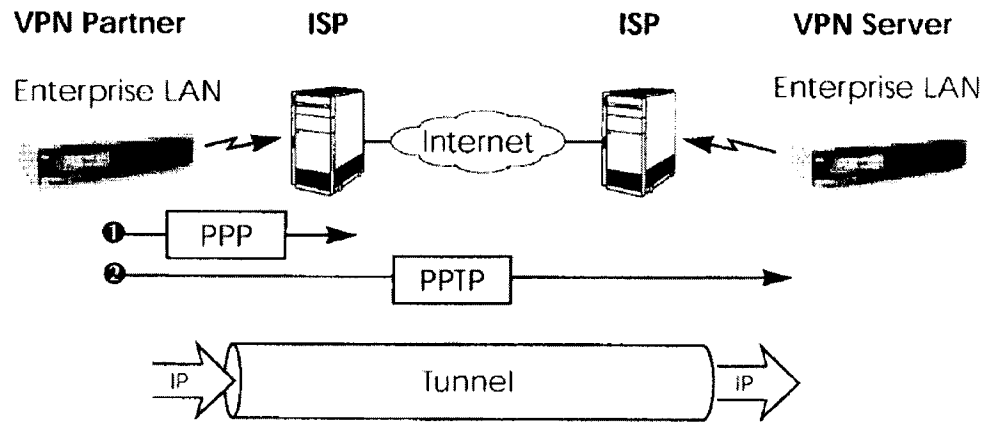
123. Even if the alleged VPN described in *BinGO EFR* is applicable to the *BinGO!* router described in *BinGO UG*, it is my opinion that *BinGO* fails to disclose the feature of a VPN between the client computer and the target computer recited in independent claim 1. *BinGO EFR* describes two scenarios for establishing an alleged VPN: a PPTP Client-to-VPN Server scenario and a LAN-to-LAN VPN scenario. (See *BinGO EFR* 83-84.) Under the PPTP Client-to-VPN Server scenario, a remote client first establishes a standard PPP connection to a local ISP, and the same client then initiates a second, logical connection, to the VPN Server. (*Id.* at 83.) A figure of this scenario is illustrated below:

### Scenario 1. PPTP Client-to-VPN Server



124. The ISP (and all intermediate Internet routers), unaware that it is participating in an alleged VPN, simply routes IP packets from the PPTP Client. (*Id.*) Under the LAN-to-LAN VPN scenario, an alleged VPN connects two enterprise LANs via the Internet, and is established via two VPN Servers. (*Id.* at 84.) A figure of this scenario is illustrated below:

## Scenario 2. LAN-to-LAN VPN



125. Either side may initiate a standard PPP link to a local ISP. (*Id.*) Once the link is established, the same server establishes a PPTP connection to the remote VPN server. (*Id.*)

126. Under either scenario, however, *BinGO EFR* describes that a connection is established to the local ISP first, and then an alleged VPN is established over the Internet. However, I understand the Request considers the user's PC as corresponding to the client computer recited in independent claim 1, and the BossPC at a corporate network as corresponding to the target computer recited in independent claim 1. (*See Req. at 190-92.*) Nowhere does *BinGO EFR* disclose that the alleged VPN under either the PPTP Client-to-VPN Server scenario or the LAN-to-LAN VPN scenario is applicable for a connection between the user's PC, which is behind the *BinGO!* router, and the BossPC, which is in the corporate network. *BinGO EFR* describes that the alleged VPN described therein requires a connection to be established with the local ISP first, and then the alleged VPN is established over the Internet. As I mentioned above, *BinGO* describes that the *BinGO!* router either connects directly to the Internet (via the ISP) or to the corporate network (without connecting to the ISP). (*See BinGO UG 15-16, Figure 1-1.*) Thus, the alleged VPN under either the PPTP Client-to-VPN Server scenario or the LAN-to-LAN VPN scenario is only described to connect to or over the Internet, and not to the corporate network at which the BossPC is located.

127. Furthermore, it is my opinion that *BinGO* has not been shown to disclose the feature of determining whether the DNS request transmitted in step (1) is requesting access to a secure web site, as recited in independent claim 1. I understand the Request contends that *BinGO's* corporate network corresponds to the secure web site of independent claim 1. (*See Req. at 192.*) The Request also appears to consider a corporate network destination as a secure destination because *BinGO*



describes that connecting to the router of the corporate network involves providing the common password to that router. (*Id.* at 185, 186, 192.) Moreover, I understand the Request appears to contend that if the alleged DNS request requests access to the corporate network destination (as is alleged under the first alleged DNS handling procedure), a determination is made that the alleged DNS request was seeking access to a secure destination, thereby disclosing the feature of determining whether the DNS request transmitted in step (1) is requesting access to a secure web site, as recited in independent claim 1. (*Id.* at 192-98.) I disagree with these contentions.

128. It is not clear, and the Request has not shown, how if the alleged DNS request requests access to the corporate network destination, a determination is made that the alleged DNS request was seeking access to a secure destination. Indeed, nowhere does the Request clearly explain how *BinGO* discloses any determination, let alone a determination of whether the alleged DNS request is requesting access to the corporate network. While the Request implies that such a determination may be done by reference to an internal table, (*id.* at 192), nowhere does the Request even refer to such a table in *BinGO* in forming its rejection of independent claim 1.

129. Furthermore, it is my opinion that *BinGO* fails to disclose automatically initiating the VPN between the client computer and the target computer in response to determining that the DNS request in step (2) is requesting access to a secure target web site. The cited portions of *BinGO* fail to disclose that the alleged VPN (e.g., the descriptions of VPNs in *BinGO* and/or the encrypted connection) between the user's PC and the BossPC is automatically initiated in response to a determination that the DNS request is requesting access to a destination.

130. Regarding the cited portions in *BinGO UG*, I understand the Request contends that page 17 of *BinGO UG* discloses automatically initiating the VPN between the client computer and the target computer in response to determining that the DNS request in step (2) is requesting access to a secure target web site. (*See* Req. at 185, 198.) I disagree. Page 17 of *BinGO UG* states:

Additionally, a significant advantage of your BinGO! is the means by which access to networks is achieved. When using a modem/ISDN-card, you must expressly dial your Internet provider in order to send an e-mail, for example. On the other hand, the router decides independently (once configured, that is) if and how a connection to the Internet provider is established. If you submit an external WWW-address with your browser, for example, your BinGO! realizes that the requested address lies outside your own LAN, thus automatically establishes a connection with your provider and the Internet. This procedure is particularly economical as your router disconnects you after a predefined time subsequent to a cessation in external data exchange.

The same principle is applicable for conveniently accessing data from your home office. While running Windows, for example, you can connect a network drive with

the server of your home office. Simply click the link in Windows Explorer and “surf” the server’s directories and files, just as you would your own hard disc.

(*Id.*) Initially, I note that nowhere does page 17 of *BinGO UG* mention anything about a VPN, let alone disclose automatically initiating the VPN between the client computer and the target computer in response to determining that the DNS request in step (2) is requesting access to a secure target web site, as recited in independent claim 1. Furthermore, I understand that the Request considers the BossPC (in a corporate network) as the target computer recited in independent claim 1, the entering of the computer name, “BossPC,” at the user’s PC as the DNS request recited in independent claim 1, and the alleged VPN as being between the user’s PC and the BossPC, as I mentioned above. However, nowhere does page 17 of *BinGO UG* disclose that a connection is established between the user’s PC and the BossPC, let alone a connection that is established between the user’s PC and the BossPC in response to entering the computer name, “BossPC.”

131. While page 17 of *BinGO UG* describes that the user may submit an external WWW-address that would make the *BinGO!* router automatically establish a connection to the provider and the Internet, nowhere does this portion of *BinGO UG* (or the rest of *BinGO*) disclose that the *BinGO!* router would automatically establish a connection between the user’s PC and the BossPC in response to the alleged DNS request (*e.g.*, entering the computer name, “BossPC,” which is not an external WWW-address). I understand the Request appears to contend that the phrase, “[t]he same principle is applicable for conveniently accessing data from your home office,” refers to the principle of automatically establishing a connection to the provider and the Internet based on the external WWW-address also being applicable to automatically establishing a connection between the user’s PC and the BossPC based on the alleged DNS request. However, this interpretation is incorrect. Initially, it is not clear what the “principle,” as recited on page 17 of *BinGO*, is referring to. Indeed, nowhere does *BinGO* define what such a principle is. For this reason alone, in my opinion, the Request has not shown that *BinGO* discloses the feature of “in response to determining that the DNS request in step (2) is requesting access to a secure target web site, automatically initiating the VPN between the client computer and the target computer,” as recited in independent claim 1.

132. Furthermore, even if the principle is referring to the concept of automatically establishing a connection to the provider and the Internet based on the external WWW-address, the phrase, “[t]he same principle is applicable for conveniently accessing data from your home office,” as recited on page 17 of *BinGO*, nevertheless is not referring to establishing a connection to a location beyond a LAN of which the user’s PC and the *BinGO!* router are a part, let alone to a

location on the corporate network (e.g., the BossPC). In particular, the phrase, “[t]he same principle is applicable for conveniently accessing data from your home office,” should be interpreted to mean (1) “the same principle is applicable for conveniently accessing data that is from your home office” rather than to mean (2) “the same principle is applicable for conveniently accessing, from your home office, data that may be at some other location.”

133. In my opinion, the former interpretation (1) is correct because page 17 of *BinGO UG* states that “you can connect a network drive with the server of your home office” so that the user can “surf the server’s directories and files.” (*Id.* (emphasis added).) Thus, to the extent that *BinGO* describes that the same principle is applicable for conveniently accessing data from the home office, *BinGO* only describes that it is applicable for accessing data that is from a server of the home office (as distinguished from accessing, from the home office, data of a server that may be at some other location).

134. Further, in my opinion, there is no indication that the server, the home office, or even the network drive, referred to on page 17 of *BinGO UG* is located outside of the LAN of which the user’s PC and the *BinGO!* router are a part. Indeed, while *BinGO UG* purposely uses the phrase “home office” on page 17, *BinGO UG* uses a different phrase (e.g., “corporate network,” “corporate head office,” “company’s head office,” etc.) to refer to a WAN partner that is outside of the LAN of which the *BinGO!* router is a part. (*Id.* at 15-16, 53.) Furthermore, *BinGO* even states that “[b]y connecting to your company’s head office from your home or branch office, you can conveniently access any information you may need from the headquarters.” (*Id.* at 15-16 (emphasis added).) Therefore, it is my opinion that one of ordinary skill in the art would understand that the home office, in addition to the server of the home office, referred to on page 17 of *BinGO UG* is in the same LAN as the user’s PC and the *BinGO!* router, and therefore is different from the corporate network. As a result, nowhere does page 17 of *BinGO UG* disclose that a connection is established between the user’s PC (in the same LAN as the *BinGO!* router) and the BossPC (at the corporate network), let alone a connection that is established between the user’s PC and the BossPC in response to the alleged DNS request.

135. I understand that if the Request considers the descriptions of alleged VPNs in *BinGO EFR* as corresponding to the VPN between the client computer and the target computer recited in independent claim 1, the Request further contends that page 82 of *BinGO EFR* discloses “in response to determining that the DNS request in step (2) is requesting access to a secure target web site, automatically initiating the VPN between the client computer and the target computer,” as recited in

independent claim 1. (See Req. at 187, 198.) Page 82 of *BinGO EFR* states, “A Virtual Private Network can be considered as a virtual Wide Area Network. It is Virtual in the sense that the network is not physical but is established on demand by software that establishes a link between a client and the server.” While *BinGO EFR* describes that the alleged VPN is established on demand by software that establishes a link between a client and the server, the Request has not shown how this alleged VPN is automatically initiated in response to anything, let alone in response to a DNS request or a determination that the DNS request is requesting access to a destination. Indeed, page 82 of *BinGO EFR* (or the rest of *BinGO*) fails to disclose when or how the software would act to establish the alleged VPN described in *BinGO EFR* to establish a link between the client and the server on demand, let alone disclose that the alleged VPN described in *BinGO EFR* is automatically initiated in response to a DNS request or any determination that the DNS request is requesting access to a destination.

136. Even assuming that the alleged VPN described in *BinGO EFR* is applicable to the *BinGO!* router, it is my opinion that the alleged VPN described in *BinGO EFR* is not automatically initiated in response to the alleged DNS request or any determination that the alleged DNS request is requesting access to a destination under the first alleged DNS handling procedure. As I mentioned above, *BinGO EFR* describes that under either a PPTP Client-to-VPN Server scenario or a LAN-to-LAN VPN scenario, a connection is established to the local ISP first, and then an alleged VPN is established from the local ISP to another VPN server over the Internet. Thus, according to *BinGO EFR*, in order for the alleged VPN to even be established, a connection to the ISP must be established first. However, under the first alleged DNS handling procedure, the only scenario in which a connection to the ISP is established is when a local DNS server does not resolve the address (i.e., because the request did not specify a secure destination), and the *BinGO!* router sends the request to a secondary DNS server (e.g., one associated with an ISP designated to be the “default route” in the *BinGO!* router configuration settings). (See Req. at 195.) But, in my opinion, such a scenario does not flow logically. *BinGO* states that the DNS server of the ISP is usually unable to translate computer names, and such a connection to the provider would be a waste of time, not to mention money. (See *BinGO UG 88*.) Thus, in this scenario in which the secondary DNS server is described to receive the alleged DNS request if the local DNS server did not resolve the address, the secondary DNS server, which is associated with the ISP according to the Request, would still be unable to resolve the address because the alleged DNS request is the entering of a computer name (e.g., “BossPC”). As a result, it is my opinion that a connection between the user’s PC and the BossPC

would not be established, let alone be established in response to the alleged DNS request or a determination that the alleged DNS request is requesting access to a destination.

137. Even if the secondary DNS server were somehow able to resolve the address, the alleged VPN would not be between the alleged client computer (the user's PC) and the alleged target computer (the BossPC at the corporate network), but rather be from the ISP associated with the secondary DNS server to the BossPC (since *BinGO EFR* describes that an alleged VPN is established from the local ISP to another VPN server over the Internet). However, nowhere does *BinGO* describe that a connection to the BossPC may be established over the Internet. As I already noted above, *BinGO* describes that the *BinGO!* router either connects directly to the Internet (via the ISP) or to the corporate network (without connecting to the ISP). (See *BinGO UG* 15-16, Figure 1-1.) Thus, the alleged VPN, which connects to the Internet from the ISP, would not even be applicable to the connection between the user's PC and the BossPC, which is at the corporate network. *BinGO* therefore does not disclose that a VPN connection between the user's PC and the BossPC would be established, let alone be established in response to the alleged DNS request or a determination that the alleged DNS request is requesting access to a destination.

138. Even if the alleged VPN between the user's PC and the BossPC is automatically initiated in response to an event, it is my opinion that the alleged VPN between the user's PC and the BossPC is nevertheless not automatically initiated in response to the alleged DNS request or any determination that the alleged DNS request is requesting access to a destination. *BinGO* describes that once a DNS request is resolved with an IP address, communication between the user's PC and the BossPC does not depend on the alleged DNS request or any determination that the alleged DNS request is requesting access to a destination. For example, *BinGO* states:

As soon as you enter www.bintec.de, for example, in the browser, the PC sends a DNS request to *BinGO!* – as *BinGO!* is known as a DNS proxy server. *BinGO!* can not translate the name itself and sends the packet with the DNS request along the default route to the provider. There the name www.bintec.de can be resolved. The DNS request is successful and in reply the PC receives the IP address for the name www.bintec.de. Now the packet can be sent on its actual journey to www.bintec.de. As *BinGO!* is entered as a gateway, and the packet has an IP address whose destination is an external LAN, the packet is sent out via the gateway (*BinGO!*).

(*BinGO UG* 90-92.) I understand that the Request contends that the same principle for resolving WWW-addresses is applicable for resolving computer names. (See Req. at 185, 191-92.) However, as I mentioned above, the same principle is not applicable for resolving computer names for a connection between the user's PC and the BossPC. Nevertheless, even assuming this were true,

*BinGO* fails to disclose, in response to determining that the DNS request in step (2) is requesting access to a secure target web site, automatically initiating the VPN between the client computer and the target computer.

139. Under the assumed scenario of *BinGO*, after sending out the alleged DNS request (e.g., the entering of the computer name “BossPC”), the user’s PC may in return receive the IP address of the BossPC. Then a packet that has the IP address may be sent out to its destination via the *BinGO!* router.

140. In this regard, the alleged VPN between the user’s PC and the BossPC is not automatically initiated in response to the alleged DNS request or any determination that the alleged DNS request is requesting access to a destination.

141. Under the second alleged DNS handling procedure of *BinGO*, I understand that the Request assumes that a client computer (e.g., the user’s PC) may query a local file (the LMHOSTS file) containing tables of entries correlating secure hostnames on a corporate network to their IP addresses. (See Req. at 195.) For example, *BinGO* states:

In the LMHOSTS file, IP addresses are arranged with their computer names in tabular form. If, for example, you are looking for BossPC, a PC located in your partner’s network (e.g., HQ), your PC asks its LMHOSTS file for the corresponding IP address and in this way is able to find the PC.

(*BinGO UG 61*.) I also understand the Request alleges that in this configuration, if the user’s PC generates a DNS request that corresponds to a secure destination (e.g., entering of the computer name “BossPC”), the user’s PC will resolve the hostname using the LMHOSTS file to obtain the IP address associated with the secure web site. (See Req. at 196.) Then, according to the Request, a connection request – in the form of an IP address – would be sent to the designated gateway (i.e., the *BinGO!* router) which, in turn, would establish the connection to the remote server. (*Id.*) For at least the reasons set forth below, it is my opinion that *BinGO* fails to disclose the features of independent claim 1 under the second alleged DNS handling procedure.

142. It is my opinion that *BinGO* fails to disclose the feature of a VPN between the client computer and the target computer recited in independent claim 1 even under the second alleged DNS handling procedure. My discussion above with respect to the first alleged DNS handling procedure applies here to the second alleged DNS handling procedure.

143. It is my opinion that *BinGO* has not been shown to disclose determining whether the DNS request transmitted in step (1) is requesting access to a secure web site, as recited in independent claim 1, even under the second alleged DNS handling procedure. My discussion above

with respect to the first alleged DNS handling procedure is equally applicable here to the second alleged DNS handling procedure.

144. It is my opinion that *BinGO* fails to disclose the feature of automatically initiating the VPN between the client computer and the target computer in response to determining that the DNS request in step (2) is requesting access to a secure target web site, as recited in independent claim 1, even under the second alleged DNS handling procedure.

145. The cited portions of *BinGO* fail to disclose that the alleged VPN between the user's PC and the BossPC is automatically initiated in response to a determination that the DNS request is requesting access to a destination. Regarding the cited portions in *BinGO UG*, my discussion above with respect to the first alleged DNS handling procedure applies to the second alleged DNS handling procedure. As I mentioned above, nowhere do the cited portions of *BinGO UG* (e.g., p. 17 of *BinGO UG*) disclose that a connection is established between the user's PC (in the same LAN as the *BinGO!* router) and the BossPC (at the corporate network), let alone a connection that is established between the user's PC and the BossPC in response to the alleged DNS request.

146. Regarding the cited portions in *BinGO EFR*, even assuming that the alleged VPN described in *BinGO EFR* is applicable to the *BinGO!* router, it is my opinion that the alleged VPN described in *BinGO EFR* is not automatically initiated in response to the alleged DNS request or any determination that the alleged DNS request is requesting access to a destination under the second alleged DNS handling procedure. As I mentioned above, *BinGO EFR* describes that under either a PPTP Client-to-VPN Server scenario and a LAN-to-LAN VPN scenario, the alleged VPN is established after the connection to the ISP is established. Thus, according to *BinGO EFR*, in order for the alleged VPN to even be established, a connection to the ISP must be established first. However, under the second alleged DNS handling procedure, the Request does not even describe that an ISP is involved. Even if the ISP were somehow involved, the alleged VPN would not be between the alleged client computer (the user's PC) and the alleged target computer (the BossPC at the corporate network), but rather be from the ISP to the BossPC (since *BinGO EFR* describes that an alleged VPN is established from the local ISP to another VPN server over the Internet). However, nowhere does *BinGO* describe that a connection to the BossPC may be established over the Internet. As I noted above, *BinGO* describes that the *BinGO!* router either connects directly to the Internet (via the ISP) or to the corporate network (without connecting to the ISP). (See *BinGO UG* 15-16 and Figure 1-1.) Thus, the alleged VPN, which connects to the Internet from the ISP, would not even be applicable to the connection between the user's PC and the BossPC, which is at the corporate

network. *BinGO* therefore does not disclose that a VPN connection between the user's PC and the BossPC would be established, let alone be established in response to the alleged DNS request or a determination that the alleged DNS request is requesting access to a destination.

147. Furthermore, even if the alleged VPN between the user's PC and the BossPC is automatically initiated in response to an event, it is my opinion that the alleged VPN between the user's PC and the BossPC is not automatically initiated in response to the DNS request or any determination that the DNS request is requesting access to a destination. My discussion above with respect to the first alleged DNS handling procedure applies to the second alleged DNS handling procedure. Accordingly, it is my opinion that *BinGO* does not disclose the feature of "in response to determining that the DNS request in step (2) is requesting access to a secure target web site, automatically initiating the VPN between the client computer and the target computer," as recited in independent claim 1.

148. Under the third alleged DNS handling procedure of *BinGO*, I understand the Request assumes that the *BinGO!* router has not been configured to have an ISP as a WAN partner, and that a corporate network is the "default" route for the *BinGO!* router. (*See* Req. at 196.) I also understand the Request alleges that all DNS requests that could not be resolved locally (i.e., computers outside of the LAN) would be routed to a DNS server on a corporate network, where the determination would be made if the request was specifying a secure destination or a non-secure destination. (*Id.*) I also understand the Request alleges that in this configuration, all DNS and Windows Internet name service (WINS) requests would be sent to the WAN partner for resolution. (*Id.* at 197.) For at least the reasons set forth below, it is my opinion that *BinGO* fails to disclose the features of independent claim 1 under the third alleged DNS handling procedure.

149. It is my opinion that *BinGO* fails to disclose the feature of a VPN between the client computer and the target computer recited in independent claim 1, even under the third alleged DNS handling procedure. My discussion above with respect to the first alleged DNS handling procedure applies equally to the third alleged DNS handling procedure.

150. It is my opinion that *BinGO* has not been shown to disclose the feature of determining whether the DNS request transmitted in step (1) is requesting access to a secure web site, as recited in independent claim 1, even under the third alleged DNS handling procedure. My discussion above with respect to the first alleged DNS handling procedure applies equally to the third alleged DNS handling procedure.

151. Furthermore, under the third alleged DNS handling procedure of *BinGO*, I understand



the Request contends that all DNS requests that could not be resolved locally are unknown packets that would be routed to the DNS server on the corporate network using the default route. (*Id.* at 196.) Because the DNS requests are unknown packets, no determination is made as to whether the DNS requests are requesting access to a secure destination or a non-secure destination..

152. It is my opinion that *BinGO* fails to disclose automatically initiating the VPN between the client computer and the target computer in response to determining that the DNS request in step (2) is requesting access to a secure target web site, as recited in independent claim 1, even under the third alleged DNS handling procedure.

153. The cited portions of *BinGO* fail to disclose that the alleged VPN between the user's PC and the BossPC is automatically initiated in response to a determination that the DNS request is requesting access to a destination. Regarding the cited portions in *BinGO UG*, my discussion above with respect to the first alleged DNS handling procedure applies to the third alleged DNS handling procedure. As I mentioned above, nowhere do the cited portions of *BinGO UG* (e.g., p. 17 of *BinGO UG*) disclose that a connection is established between the user's PC (in the same LAN as the *BinGO!* router) and the BossPC (at the corporate network), let alone a connection that is established between the user's PC and the BossPC in response to the alleged DNS request.

154. Regarding the cited portions in *BinGO EFR*, even assuming that the alleged VPN described in *BinGO EFR* is applicable to the *BinGO!* router, the alleged VPN described in *BinGO EFR* is not automatically initiated in response to the alleged DNS request or any determination that the alleged DNS request is requesting access to a destination under the third alleged DNS handling procedure. As I mentioned above, *BinGO EFR* describes that under either a PPTP Client-to-VPN Server scenario or a LAN-to-LAN VPN scenario, the alleged VPN is only established after the connection to the ISP is established. Thus, according to *BinGO EFR*, in order for the alleged VPN to even be established, a connection to the ISP must be established first. However, under the third alleged DNS handling procedure, the Request does not even describe that an ISP is involved. Even if the ISP were somehow involved, the alleged VPN would not be between the alleged client computer (the user's PC) and the alleged target computer (the BossPC at the corporate network), but rather be from the ISP to the BossPC (since *BinGO EFR* describes that an alleged VPN is established from the local ISP to another VPN server over the Internet). However, nowhere does *BinGO* describe that a connection to the BossPC may be established over the Internet. As I already noted above, *BinGO* describes that the *BinGO!* router either connects directly to the Internet (via the ISP) or to the corporate network (without connecting to the ISP). (*See BinGO UG* 15-16, Figure 1-1.) Thus, the

alleged VPN, which connects to the Internet from the ISP, would not even be applicable to the connection between the user's PC and the BossPC, which is at the corporate network. *BinGO* therefore does not disclose that a VPN connection between the user's PC and the BossPC would be established, let alone be established in response to the alleged DNS request or a determination that the alleged DNS request is requesting access to a destination.

155. Furthermore, even if the alleged VPN between the user's PC and the BossPC is automatically initiated in response to an event, the alleged VPN between the user's PC and the BossPC is not automatically initiated in response to any determination that the alleged DNS request is requesting access to a destination. Under the third alleged DNS handling procedure of *BinGO*, I understand the Request concedes that all DNS requests that could not be resolved locally would be routed to a DNS server on a corporate network using a default route. (*See* Req. at 196.) The BossPC is located at this corporate network. However, I understand the Request concedes that the determination (of whether the DNS request was specifying a secure destination or a non-secure destination) is made at the corporate network. (*Id.*) In this regard, based on the configuration assumed by the Request, the alleged DNS request is routed to the corporate network before any determination is made as to whether the alleged DNS request is specifying a secure destination or a non-secure destination.

156. Indeed, *BinGO* states that the "default route leads all unknown packets to your head office." (*See BinGO UG 90.*) Under the third alleged DNS handling procedure of *BinGO*, I understand the Request contends that all DNS requests that could not be resolved locally are unknown packets that would be routed to the DNS server on the corporate network using the default route. (*See* Req. at 196.) Because the DNS requests are unknown packets, the DNS requests would be forwarded as unknown packets to the corporate network using the default route before any determination is made as to whether the DNS requests are requesting access to a secure destination or a non-secure destination. As a result, the alleged VPN between the user's PC and the BossPC is not automatically initiated in response to any determination that the alleged DNS request is requesting access to a destination.

157. Regarding dependent claim 2, I understand that under the second alleged DNS handling procedure, the Request assumes that a client computer (e.g., the user's PC) may query a local file (the LMHOSTS file) containing tables of entries correlating secure hostnames on a corporate network to their IP addresses. (*See* Req. at 195.) I understand the Request alleges that in this configuration, if the user's PC generates a DNS request that corresponds to a secure destination

(e.g., entering of the computer name “BossPC”), the user’s PC will resolve the hostname using the LMHOSTS file to obtain the IP address associated with the secure web site. (*Id.* at 196.) Then, according to the Request, a connection request – in the form of an IP address – would be sent to the designated gateway (i.e., the BinGO! router) which, in turn, would establish the connection to the remote server. (*Id.*) Thus, the Request concedes that under the second alleged DNS handling procedure, the alleged determining of BinGO (e.g., requesting access to the BossPC) is performed at the alleged client computer (e.g., the user’s PC). Accordingly, it is my opinion that BinGO fails to disclose that steps (2) and (3) are performed at a DNS server separate from the client computer.

158. Regarding dependent claim 3, I understand that under each of the first, second, and third alleged DNS handling procedures, the Request considers entering the computer name, “BossPC,” in the user’s PC as the DNS request recited in independent claim 1, the corporate network of *BinGO* as the secure web site recited in independent claim 1, and the Internet destination of *BinGO* as corresponding to a non-secure web site recited in claim 3. (*See* Req. at 190-98 and 199-200.) However, even assuming these contentions were true, under none of the first, second, and third alleged DNS handling procedures would the *BinGO!* router determine that the alleged DNS request (e.g., entering the computer name “BossPC”) is not requesting access to the alleged secure web site (e.g., the corporate network). In other words, if the alleged DNS request of *BinGO* is entering the computer name, “BossPC,” then it is my opinion that *BinGO* necessarily describes that the DNS request is requesting access to the corporate network, which is where the BossPC is located.

159. Regarding dependent claim 4, it is my opinion that *BinGO* fails to disclose “returning an error from the DNS request,” as recited in claim 4. Claim 4 recites the feature of “determining whether the client computer is authorized to establish a VPN with the target computer and, if not so authorized, returning an error from the DNS request.” However, I understand the Request asserts that the alleged error returned to a remote user for failing to log in and authenticate to the *BinGO!* router corresponds to the feature of “returning an error from the DNS request.” (*See* Req. at 201.) I disagree.

160. First, the alleged error that the Request refers to is in connection with a user failing to log in to the *BinGO!* router. However, logging in to the *BinGO!* router has nothing to do with determining whether the client computer is authorized to establish a VPN with the target computer. Logging in to the *BinGO!* router simply allows a user to configure the *BinGO!* router.

161. Secondly, the Request makes this assertion without providing any support. Nowhere does *BinGO* disclose returning an error, let alone returning an error from the DNS request. If the

Request considers this feature as inherently disclosed in *BinGO*, such a contention would be incorrect. It is my opinion that one of ordinary skill in the art can ascertain that an error is not necessarily returned if a router, for example, determines that a client computer is not authorized to establish a VPN with a target computer. As I mentioned above, there is no mention of returning an error in *BinGO*, let alone any returning an error from the DNS request.

162. Regarding dependent claim 5, I understand that according to the Request, *BinGO UG* shows that the *BinGO!* router could be configured to route all DNS requests to a corporate network gateway or router for resolution, citing support at page 90 of *BinGO UG*. (*Id.* at 218.) I also understand that according to the Request, *BinGO UG* also explains that a *BinGO!* router would have to authenticate itself with the destination router in the corporate network before it would transfer data to the corporate network, citing support at page 40 of *BinGO UG*. (*Id.*) I also understand the Request appears to contend that when the *BinGO!* router is configured to send all DNS traffic to the corporate network for IP resolution, authentication would be required before the DNS request – regardless of whether it specified a secure or non-secure destination – would be resolved. (*Id.*) Even assuming that this scenario were true, *BinGO* nevertheless fails to disclose, prior to automatically initiating the VPN between the client computer and the target computer, determining whether the client computer is authorized to resolve addresses of non-secure target computers and, if not so authorized, returning an error from the DNS request,.

163. Initially, it is my opinion that *BinGO* fails to disclose determining whether the client computer is authorized to resolve addresses. None of the cited portions of *BinGO* determining whether the alleged client computer (*e.g.*, the user's PC) is authorized to resolve addresses of any computers, let alone addresses of non secure target computers. *BinGO* states:

Before every connection, *BinGO!* and the router at HQ check the incoming data to see if they should take the call. In order to protect the network against unauthorized access, acceptance of the call only takes place after correct authentication. This authentication is based on a common password and two codes that you and your partner use for the connection.

(*BinGO UG* 40.) Thus, the authentication in *BinGO* that is referred to by the Request is for authenticating against unauthorized access to the network between the *BinGO!* router and the router at HQ. However, determining if the user's PC is authorized to access a network is not the same as determining if the user's PC is authorized to resolve addresses not located at that network. Indeed, nowhere does *BinGO* disclose that any determination is made as to whether the user's PC is authorized to resolve addresses of any computer, let alone addresses of non-secure target computers.

164. Even if authenticating against unauthorized access corresponds to determining if the user's PC is authorized to resolve addresses, it is my opinion that *BinGO* nevertheless fails to disclose that a determination is made as to whether the user's PC is authorized to resolve addresses of non secure target computers. As I mentioned above, I understand the Request considers entering the computer name, "BossPC," as corresponding to the DNS request recited in independent claim 1, and the corporate network as corresponding to the secure web site recited in independent claim 1. Furthermore, I understand that, according to the Request, the computer name "BossPC," is routed to the corporate network to resolve an address for this alleged DNS request. Thus, it does not logically follow why *BinGO*, in such a scenario, would still describe that a determination is made as to whether the user's PC is authorized to resolve addresses of non-secure target computers, especially since the alleged DNS request is supposedly a request to access the corporate network (which the Request considers as corresponding to a secure destination). Indeed, the Request even concedes that there is no relation between the authentication of access to the network (between the *BinGO!* router and the router at HQ) and whether or not the destination is secure or non-secure. (See Req. at 218 ("authentication would be required before the DNS request – regardless of whether it specified a secure or non-secure destination – would be resolve") (emphasis added).).

165. Furthermore, it is my opinion that *BinGO* fails to disclose the feature of determining whether the client computer is authorized to resolve addresses prior to automatically initiating the VPN. Even if the authentication of access to the network between the *BinGO!* router and the router at HQ can be considered as corresponding to the feature of "determining whether the client computer is authorized to resolve addresses," *BinGO* nevertheless fails to disclose that such an authentication occurs prior to the alleged VPN between the user's PC and the BossPC is initiated. As stated in *BinGO*, the *BinGO!* router and the router at HQ check the incoming data (e.g., the common password) to see if they should take the call, and acceptance of the call only takes place after correct authentication. (See *BinGO UG 40*.) Thus, *BinGO* describes that a connection between the *BinGO!* router and the router at HQ is established (in order to check the incoming data) before authentication occurs. In this regard, the network between the *BinGO!* router and the router at HQ (e.g., the router at the corporate network) would already be initiated.

166. Furthermore, it is my opinion that *BinGO* fails to disclose returning an error from the DNS request. I understand the Request fails to cite to a teaching in *BinGO* that corresponds to the feature of returning an error from the DNS request if the client computer is not authorized to resolve addresses of non secure target computers. Indeed, the Request fails to mention anything about an

error or returning an error, let alone discuss the feature of returning an error from the DNS request if the client computer is not authorized to resolve addresses of non secure target computers.

167. Regarding dependent claim 6, I understand the Request contends that the NAT procedure described in pages 167-168 and 244-246 of *BinGO UG* corresponds to the IP hopping scheme recited in claim 6. (*See* Req. at 202.) Alternatively, I understand the Request contends that the open shortest path first (OSPF) protocol described in page 17 of *BinGO EFR* corresponds to the IP hopping scheme recited in claim 6. (*Id.* at 202-203.)

168. Regarding the NAT procedure, as an example of an IP hopping scheme, the '135 patent describes that a pair of nodes may agree upon an algorithm for hopping between IP addresses (both sending and receiving) such that an eavesdropper sees apparently continuously random IP address pairs (source and destination) for packets transmitted between the pair. (*See* '135 patent at 5:52-57.) However, the description of NAT in the cited portion of *BinGO UG* mentions nothing about hopping between IP addresses, an agreed upon algorithm for hopping between IP addresses, or even apparently continuously random IP address pairs. For example, *BinGO UG* states:

► ► NAT is a simple-to-operate procedure that can be used for four purposes in the BinTec implementation:

- Hiding the internal host addresses of a LAN by remapping to one or more external addresses. The external addresses remain unchanged.
- Controlling the external to internal access. Externally the router forwards all ► ► data packets, internally it only forwards what has been explicitly enabled (Forward NAT).
- Reverse NAT ensures that a connection partner uses only a single ► ► IP address. Only incoming connections are allowed from the partner, *e.g.*, as a service from Internet Service Providers (ISP).
- Permanent monitoring of the connections into and out of a network via the router with indication of the source and destination addresses and ► ► ports.

NAT always refers to an interface. The LAN to which *BinGO!* is connected is always referred to as "inside", the WAN partner as "outside".

(*See BinGO UG* 244-46.) I understand the Request simply points to this description of NAT in *BinGO UG* without clearly explaining how the NAT in *BinGO* corresponds to an IP hopping scheme. Rather, the Request alleges that the NAT referred to in *BinGO UG* "is used to route IP packets between client and destination computers." (*See* Req. at 202). Nowhere does the cited portion of *BinGO UG* mention anything about "IP packets" or how such packets correspond to an IP hopping

scheme. Furthermore, the NAT is not an IP hopping scheme. Indeed, even RFC 2663, submitted by the Requester as Exhibit Y17, states, “NAT devices attempt to provide a transparent routing solution to end hosts trying to communicate from disparate address realms,” and that “[t]his solution only works when the applications do not use the IP addresses as part of the protocol itself.” (See RFC 2663 at 1-2 (emphasis added)).

169. Regarding the OSPF protocol, the description of OSPF protocol in the cited portion of *BinGO EFR* mentions nothing about hopping between IP addresses, an agreed upon algorithm for hopping between IP addresses, or even apparently continuously random IP address pairs. For example, *BinGO EFR* discloses that OSPF is an interior routing protocol that is often used by larger network installations as an alternative to routing information protocol (RIP). (See *BinGO EFR* 17.) *BinGO EFR* discloses that one problem with RIP that OSPF addresses includes no hop-count limitations. (*Id.*) However, I understand the Request simply points to the description of OSPF on page 17 of *BinGO EFR* without explaining how OSPF corresponds to an IP hopping scheme. Indeed, nowhere does page 17 of *BinGO EFR* even mention an IP address, let alone an IP address hopping scheme.

170. Furthermore, it is my opinion that the OSPF described in *BinGO EFR* is not applicable to the *BinGO!* router described in *BinGO! UG*. *BinGO EFR* states that “[d]epending on your particular product some of the features described in this document may not be available on your system.” (See *BinGO EFR* 2.) There is no indication that the OSPF described in *BinGO EFR* is even applicable to the *BinGO!* router described in *BinGO UG*. Indeed, nowhere in the entire chapter of *BinGO EFR* describing OSPF (*e.g.*, pages 6-42) does *BinGO EFR* mention that the described OSPF is available for the *BinGO!* router. Furthermore, one of ordinary skill in the art would understand that the *BinGO!* router would not use the OSPF protocol because the OSPF protocol is typically reserved for large enterprise networks and ISPs.

171. Regarding dependent claim 7, I understand the Request considers the alleged VPN under the LAN-to-LAN VPN scenario in *BinGO EFR* as corresponding to the VPN recited in claim 7. (See Req. at 203-04.) However, under the LAN-to-LAN VPN scenario, an alleged VPN connects two enterprise LANs via the Internet, *and* is established via two VPN Servers. (See *BinGO EFR* 84.) As I mentioned above with respect to independent claim 1, the alleged VPN under the LAN-to-LAN VPN scenario is not applicable to the *BinGO!* router described in *BinGO UG*. While I understand the Request considers the *BinGO!* router as corresponding to the gatekeeper computer recited in independent claim 7, nowhere does *BinGO EFR* describe that the *BinGO!* router is used to connect

the two enterprise LANS together.

172. Regarding dependent claim 8, it is my opinion that *BinGO* fails to disclose the feature in which “step (2) is performed in a DNS proxy server that passes through the request to a DNS server if it is determined in step (3) that access is not being requested to a secure target web site,” as recited in claim 8. As I noted above with respect to independent claim 1, under each of the first, second, and third alleged DNS handling procedures, I understand the Request considers entering the computer name, “BossPC,” in the user’s PC as corresponding to the DNS request recited in independent claim 1, the corporate network of *BinGO* as corresponding to the secure web site recited in independent claim 1, and the Internet destination of *BinGO* as corresponding to a non-secure web site recited in claim 3. (*See* Req. at 190-200.) However, even assuming these contentions were true, under none of the first, second, and third alleged DNS handling procedures would the *BinGO!* router determine that the alleged DNS request (*e.g.*, entering the computer name “BossPC”) is not requesting access to the alleged secure web site (*e.g.*, the corporate network). In other words, if the alleged DNS request of *BinGO* is entering the computer name, “BossPC,” then *BinGO* necessarily describes that the DNS request is requesting access to the corporate network, which is where the BossPC is located.

173. Moreover, as I mentioned above with respect to independent claim 1, I understand the Request appears to contend that if the alleged DNS request requests access to the corporate network destination rather than the Internet destination, a determination is made that the alleged DNS request was seeking access to a secure destination. Even if this alleged determination also corresponds to determining that access is not being requested to a secure target web site, it is my opinion that this alleged determination is not performed in a DNS proxy server under each of the three different alleged DNS handling procedures.

174. I understand the Request appears to contend that the *BinGO!* router corresponds to the DNS proxy server recited in claim 8. (*Id.* at 194-96, 204-05.) However, under the first alleged DNS handling procedure, the alleged determination occurs in the local DNS server (that resolves the alleged DNS request) rather than the *BinGO!* router. Under the second alleged DNS handling procedure, the alleged determination occurs in the user’s PC (that contains the LMHOSTS file) rather than the *BinGO!* router. Under the third alleged DNS handling procedure, I understand the Request contends that all DNS requests that could not be resolved locally are unknown packets that would be routed to the DNS server on the corporate network using the default route. (*Id.* at 196.) Because the DNS requests are unknown packets, no determination is made as to whether the DNS requests are



requesting access to a secure destination or a non-secure destination. Correspondingly, under the third alleged DNS handling procedure, it is my opinion that *BinGO* does not disclose any determination, let alone any determination performed in the alleged DNS proxy server (e.g., the *BinGO!* router). In view of the foregoing, it is my opinion that *BinGO* fails to disclose the feature in which “step (2) is performed in a DNS proxy server that passes through the request to a DNS server if it is determined in step (3) that access is not being requested to a secure target web site,” as recited in claim 8.

175. Regarding dependent claim 9, I understand the Request contends that the *BinGO!* router prompting a login and password to a user corresponds to the feature of “transmitting a message to the client computer to determine whether the client computer is authorized to establish the VPN target computer.” (*Id.* at 205). However, as I mentioned above, logging in to the *BinGO!* simply allows a user to configure the *BinGO!* router, which is different from determining whether the user’s PC is authorized to establish a VPN, let alone a VPN target computer.

176. I also understand the Request points to description in page 242 of *BinGO UG* regarding a point-to-point protocol (PPP) as corresponding to “transmitting a message to the client computer to determine whether the client computer is authorized to establish the VPN target computer.” (*Id.* at 205). Page 242 of *BinGO UG* describes that challenge handshake authentication protocol (CHAP) and MS-CHAP are common procedures used for authentication of PPP connections, and that these protocols use a standard procedure to exchange a user ID and a password for checking the identity of the far end. (*Id.*) However, the cited portion of *BinGO UG* does not mention anything about a VPN, let alone a VPN target computer. I understand the Request points to *BinGO EFR* as allegedly disclosing a VPN in the rejection of independent claim 1. I also understand the Request fails to even point to *BinGO EFR* in the rejection of claim 9. Nevertheless, *BinGO EFR* makes no mention of transmitting a message to the user’s PC to determine whether the user’s PC is authorized to establish a VPN, let alone a VPN target computer. In view of the foregoing, it is my opinion that *BinGO* fails to disclose the feature in which “step (3) comprises the step of transmitting a message to the client computer to determine whether the client computer is authorized to establish the VPN target computer,” as recited in claim 9.

177. Regarding independent claim 10, I understand the Request contends that a user’s PC corresponds to the client computer recited in independent claim 10, that entering the computer name, “BossPC,” at the user’s PC corresponds to the request recited in independent claim 10, that the actual BossPC itself corresponds to the secure target computer recited in independent claim 10, that the

*BinGO!* router corresponds to the gatekeeper computer recited in independent claim 10, and that the user's PC or the *BinGO!* router corresponds to the DNS proxy server recited in independent claim 10. (See Req. at 206-08.) I also understand the Request contends that *BinGO* describes three different DNS handling procedures, each of which allegedly corresponds to the feature of determining if a DNS request is specifying a secure destination. (*Id.* at 194-98, 206-07.)

178. If the Request considers the description of alleged VPNs in *BinGO EFR* (e.g., either under the PPTP Client-to-VPN Server scenario and the LAN-to-LAN VPN scenario) as corresponding to the VPN recited in independent claim 10, it is my opinion that *BinGO* fails to disclose a gatekeeper computer that allocates resources for the VPN between the client computer and the secure web computer in response to the request by the DNS proxy server, as recited in independent claim 10.

179. Under the PPTP Client-to-VPN Server scenario, a remote client first establishes a standard PPP connection to a local ISP, and the same client then initiates a second, logical connection, to the VPN Server. (See *BinGO EFR* 83.) As I mentioned above with respect to independent claim 1, the alleged VPN under the PPTP Client-to-VPN Server scenario is not applicable to the *BinGO!* router described in *BinGO UG*. While I understand the Request considers the *BinGO!* router as corresponding to the gatekeeper computer recited in independent claim 10, nowhere does *BinGO EFR* describe that any router, let alone the *BinGO!* router, is used to connect the remote client to the local ISP. Under the PPTP Client-to-VPN Server scenario, the remote client initiates the alleged VPN by establishing a direct connection to the local ISP first without the use of a router. (*Id.*) Thus, it is my opinion that *BinGO* fails to disclose the feature of "a gatekeeper computer that allocates resources for the VPN between the client computer and the secure web computer in response to the request by the DNS proxy server," as recited in independent claim 10.

180. Under the LAN-to-LAN VPN scenario, an alleged VPN connects two enterprise LANs via the Internet, and is established via two VPN Servers. (*Id.* at 84.) As I mentioned above with respect to independent claim 1, the alleged VPN under the LAN-to-LAN VPN scenario is not applicable to the *BinGO!* router described in *BinGO UG*. While I understand the Request considers the *BinGO!* router as corresponding to the gatekeeper computer recited in independent claim 10, nowhere does *BinGO EFR* describe that the *BinGO!* router is used to connect the two enterprise LANS together.

181. Regarding dependent claim 12, if the Request considers the description of alleged VPNs in *BinGO EFR* (e.g., either under the PPTP Client-to-VPN Server scenario and the LAN-to-

LAN VPN scenario) as corresponding to the VPN recited in claim 12, it is my opinion that *BinGO* nevertheless fails to disclose the feature in which the gatekeeper computer determines whether the client computer has sufficient security privileges to create the VPN and, if the client computer lacks sufficient security privileges, rejecting the request to create the VPN, as recited in claim 12.

182. Under the PPTP Client-to-VPN Server scenario, a remote client first establishes a standard PPP connection to a local ISP, and the same client then initiates a second, logical connection, to the VPN Server. (*See BinGO EFR 83.*) As I mentioned above with respect to independent claim 1, the alleged VPN under the PPTP Client-to-VPN Server scenario is not applicable to the *BinGO!* router described in *BinGO UG*. While I understand the Request considers the *BinGO!* router as corresponding to the gatekeeper computer recited in claim 12, nowhere does *BinGO EFR* describe that any router, let alone the *BinGO!* router, determines whether the user's PC has sufficient security privileges to create the alleged VPN under the PPTP Client-to-VPN server scenario. Indeed, under the PPTP Client-to-VPN Server scenario, the remote client initiates the alleged VPN by establishing a direct connection to the local ISP first without the use of a router. (*Id.*)

183. Under the LAN-to-LAN VPN scenario, an alleged VPN connects two enterprise LANs via the Internet, and is established via two VPN Servers. (*Id.* at 84.) As I mentioned above with respect to independent claim 1, the alleged VPN under the LAN-to-LAN VPN scenario is not applicable to the *BinGO!* router described in *BinGO UG*. While I understand the Request considers the *BinGO!* router as corresponding to the gatekeeper computer recited in claim 12, nowhere does *BinGO EFR* describe that the *BinGO!* router determines whether the user's PC has sufficient security privileges to create the alleged VPN under the LAN-to-LAN VPN scenario.

184. Regarding independent claim 13, *BinGO* describes that a user may log in to the *BinGO!* router with a user name and password to configure the *BinGO!* router. (*See BinGO UG 13-14.*) *BinGO* also describes that there are several ways of restricting logging in and access to the *BinGO!* router to authorized users only. (*Id.* at 240.) For example, *BinGO* states, "You can log in to *BinGO!* in several different ways ... but logging in is always protected by a password." (*Id.*) *BinGO* also states that additional user accounts may be created, and that a certain password and a certain action can be assigned to a user. (*Id.* at 241.)

185. *BinGO* also describes that the *BinGO!* router may be used as a Dynamic Host Configuration Protocol (DHCP) server:

Every PC in your LAN requires its own IP address, just as *BinGO!* does. Otherwise the devices could not communicate together.

...

These IP addresses can be fixed on PCs. The disadvantage is that if you are newly configuring or reconfiguring your network, you have to assign each PC individually with its own IP address. This can involve quite a lot of work depending on the number of PCs you have on your network.

You can save yourself all this work with a ► ► DHCP server (DHCP=Dynamic Host Configuration Protocol). Automatically, a DHCP server allocates IP addresses to all the PCs on the LAN. The PCs are then DHCP clients. All you have to do is define a pool of IP addresses that the DHCP server may allocate to computers on the network. In addition, you must tell the PCs that they should request their IP addresses from the server.

(*Id.* at 84, Figure 4-2.) *BinGO* also describes that every PC that newly enters the network – after booting, for example – sends out an address request and in reply, receives its IP address. (*Id.* at 85.) Usually, the PC retains this address for a specified period of time. (*Id.*) Afterwards, the address is reassigned. (*Id.*)

186. *BinGO* also describes authentication for a connection between the *BinGO!* router and a router at HQ (*e.g.*, a router at the corporate network):

In order to connect with a WAN partner (*e.g.*, corporate head office), you will need some pieces of information about the remote terminal that should take your call. Likewise, the remote terminal must know information about you. This data must be commonly agreed upon by the equipment on both sides of the connection.

Before every connection, *BinGO!* and the router at HQ check the incoming data to see if they should take the call. In order to protect the network against unauthorized access, acceptance of the call only takes place after correct authentication. This authentication is based on a common password and two codes that you and your partner use for the connection.

(*Id.* at 40.)

187. I understand the Request contends that (1) the user's PC corresponds to the authorized client recited in independent claim 13, that (2) the *BinGO!* router corresponds to the central computer recited in independent claim 13, that (3) a computer on the corporate network of *BinGO* (*e.g.*, BossPC) corresponds to the second computer recited in independent claim 13, that (4) a DNS request from the user's PC corresponds to the request to establish a connection recited in independent claim 13, that (5) the encrypted connection between the user's PC and the BossPC corresponds to the virtual private link recited in independent claim 13, and that (6) the *BinGO!* router allowing a user to route traffic through it to a specified destination corresponds to the feature of allocating resources recited in independent claim 13. (*See Req.* at 208-211.) It is unclear what

teaching of *BinGO* the Request considers as corresponding to the plurality of authentication tables recited in independent claim 13.

188. For example, I understand the Request appears to point to three different embodiments of *BinGO* as allegedly corresponding to the plurality of authentication tables recited in independent claim 13. First, it appears that the Request is contending that the *BinGO!* router may be used as a DHCP server, and that the LAN IP addresses (assigned to PCs in the same LAN as the *BinGO!* router) correspond to the plurality of authentication tables recited in independent claim 13. (*Id.* at 210-211.) Second, it appears that the Request is also contending that the password for logging in to the *BinGO!* router for a user corresponds to the plurality of authentication tables as recited in independent claim 13. (*Id.* at 210.) Third, it appears that the Request is also contending that the local name of the user's PC (*e.g.*, LittleIndian) and the common password between the user's PC and the WAN partner (*e.g.*, the corporate network) correspond to the plurality of authentication tables as recited in independent claim 13. (*Id.*)

189. Even though the foregoing three embodiments of *BinGO* are different from one another, I understand that the Request mixes and matches features from these three embodiments together. For example, I understand the Request contends that the router at HQ (*e.g.*, router at the corporate network) checking incoming data (*e.g.*, the common password) to see if it should take a call from the *BinGO!* router for authentication corresponds to the feature of allocating resources to establish a virtual private link between the client and a second computer responsive to a determination that the request is from an authorized client. (*Id.* at 211.) The checking of the incoming data described in *BinGO*, however, corresponds to the third embodiment (*e.g.*, use of the local name of the user's PC and the common password between the user's PC and the corporate network), and not the first two embodiments referred to in the Request.

190. Nevertheless, even assuming that the Request considers the three embodiments of *BinGO* as separately corresponding to the plurality of authentication tables recited in independent claim 13, it is my opinion that *BinGO* nevertheless fails to disclose the features of independent claim 13, for at least the following reasons.

191. Under the first embodiment of *BinGO*, the *BinGO!* router may serve as a DHCP server that assigns LAN IP addresses to PCs (*e.g.*, the user's PC) in the same LAN as the *BinGO!* router. (*See Req.* at 210-211.) I understand the Request contends that the LAN IP addresses correspond to the plurality of authentication tables recited in independent claim 13. Even assuming that this were true, it is my opinion that *BinGO* nevertheless fails to disclose the features of

independent claim 13.

192. Initially, I understand the Request contends that authenticating log in access to the *BinGO!* router corresponds to the feature of “authenticating, with reference to one of the plurality of authentication tables, that the request received in step (1) is from an authorized client,” as recited in independent claim 13. (*Id.* at 210-211.) However, nowhere does *BinGO* disclose that authenticating log in access to the *BinGO!* router (the alleged authenticating) is related to the assigned LAN IP addresses (the alleged plurality of authentication tables) or even the DNS request from the user’s PC (the alleged request), let alone disclose authenticating log in access to the *BinGO!* router, with reference to one of the assigned LAN IP addresses, that the DNS request is from an authorized client. Indeed, I understand that the Request fails to disclose how these three different concepts are even related. Thus, it is my opinion that authenticating log in access to the *BinGO!* router, the LAN IP addresses assigned by the *BinGO!* router serving as a DHCP server, and the DNS request from the user’s PC are all concepts that are independent of one another.

193. For example, *BinGO* describes that a user may log in to the *BinGO!* router with a user name and password to configure the *BinGO!* router. (*See BinGO UG* 13-14.) Thus, the user may log in to the *BinGO!* router not by referring to the LAN IP address of the user’s PC, but simply by referring to the user’s name and password. Furthermore, the user is seen to log in to the *BinGO!* router regardless of any DNS request that is generated from the user’s PC. Thus, nowhere does *BinGO* disclose that authenticating log in access to the *BinGO!* router (the alleged authenticating) is related to the assigned LAN IP addresses (the alleged plurality of authentication tables) and/or the DNS request from the user’s PC (the alleged request), let alone disclose authenticating log in access to the *BinGO!* router, with reference to one of the assigned LAN IP addresses, that the DNS request is from an authorized client.

194. Even if the Request considers the assigning of the LAN IP addresses by the *BinGO!* router serving as a DHCP server (rather than authenticating log in access to the *BinGO!* router) as corresponding to the authenticating recited in independent claim 13, it is my opinion that *BinGO* nevertheless fails to disclose the authenticating recited in independent claim 13. As I mentioned above, if the *BinGO!* router is used as a DHCP server, the *BinGO!* router merely assigns PCs (in the same LAN as the *BinGO!* router) IP addresses. (*See BinGO UG* 84-85.) Assigning an IP address to a PC, however, is not the same as authenticating the PC or authenticating that a request from the PC is from an authorized client. Furthermore, nowhere does *BinGO* disclose that authentication is needed in order for the *BinGO!* router to assign an IP address to a PC connected to the *BinGO!*

router. Thus, it is my opinion that *BinGO* does not disclose any form of authenticating with respect to the *BinGO!* router serving as a DHCP server, let alone disclose authenticating, with reference to one of the assigned LAN IP address, that the DNS request from that PC is from an authorized client. Furthermore, nowhere does *BinGO* disclose any relation between (1) assigning the IP address to the user's PC and (2) a DNS request from the user's PC. In other words, the *BinGO!* router assigns the IP address to the user's PC regardless of the DNS request from the user's PC or any authentication that the DNS request is received from an authorized client.

195. Furthermore, it is my opinion that *BinGO* fails to disclose, responsive to a determination that the request is from an authorized client, allocating resources to establish a virtual private link between the client and a second computer, as recited in independent claim 13. Nowhere does *BinGO* disclose that the *BinGO!* router allowing a user to route traffic through it to a specified destination (the alleged allocating resources) to establish the encrypted connection between the user's PC and the BossPC (the alleged virtual private link) is in response to the DNS request from the user's PC (the alleged request), let alone in response to a determination that the DNS request from the user's PC is from an authorized client. For example, as I mentioned above with respect to independent claim 1, *BinGO* fails to disclose that an alleged VPN between the user's PC and the BossPC is automatically initiated in response to any DNS request. For at least the same reasons as independent claim 1, it is my opinion that *BinGO* fails to disclose that the *BinGO!* router allowing a user to route traffic through it to a specified destination to establish the encrypted connection between the user's PC and the BossPC is in response to the DNS request from the user's PC, let alone in response to a determination that the DNS request from the user's PC is from an authorized client.

196. Under the second embodiment of *BinGO*, I understand the Request contends that the password for logging in to the *BinGO!* router for a user corresponds to the plurality of authentication tables as recited in independent claim 13. (*Id.* at 210.) I disagree for the following reasons.

197. Even if the password for logging in to the *BinGO!* router may be considered as corresponding to the plurality of authentication tables recited in independent claim 13, it is my opinion that *BinGO* nevertheless fails to disclose authenticating, with reference to one of the plurality of authentication tables, that the request received in step (1) is from an authorized client, as recited in independent claim 13. As I mentioned above, a user may log in to the *BinGO!* router to configure one or more settings of the *BinGO!* router. (*See BinGO UG 13-14.*) Thus, logging in to the *BinGO!* router with the password simply allows the user to configure one or more settings of the *BinGO!*

router. I understand the Request contends that authenticating log in access to the *BinGO!* router corresponds to the authenticating recited in independent claim 13. (*See* Req. at 210-211.) However, nowhere does *BinGO* disclose any relation between the password for logging in to the *BinGO!* router (the alleged plurality of authentication tables) and the DNS request from the user's PC (the alleged request). I understand the Request even fails to disclose how these two concepts are even related. Indeed, logging in to the *BinGO!* router and authenticating based on that password is independent of the DNS request received from the user's PC.

198. For example, regardless of whether the user can log in to the *BinGO!* router, the *BinGO!* router is nevertheless able to receive DNS requests from the user's PC (assuming that the *BinGO!* router is already configured to communicate with the user's PC). On the other hand, if the *BinGO!* router is not already configured to communicate with the user's PC, then the DNS requests would not even be received by the *BinGO!* router in the first place. Thus, *BinGO* fails to disclose how logging in to the *BinGO!* router and authenticating with reference to the password is related in any way to the DNS request, let alone disclose authenticating, with reference to the password for logging in to the *BinGO!* router, that the DNS request is received from an authorized client. Furthermore, it is my opinion that *BinGO* fails to disclose the feature of responsive to a determination that the request is from an authorized client, allocating resources to establish a virtual private link between the client and a second computer, as recited in independent claim 13. Nowhere does *BinGO* disclose that the *BinGO!* router allowing a user to route traffic through it to a specified destination (the alleged allocating resources) to establish the encrypted connection between the user's PC and the BossPC (the alleged virtual private link) is in response to the DNS request from the user's PC (the alleged request), let alone in response to a determination that the DNS request from the user's PC is from an authorized client. For example, as I mentioned above with respect to independent claim 1, *BinGO* fails to disclose that an alleged VPN between the user's PC and the BossPC is automatically initiated in response to any DNS request. For at least the same reasons as independent claim 1, *BinGO* fails to disclose that the *BinGO!* router allowing a user to route traffic through it to a specified destination to establish the encrypted connection between the user's PC and the BossPC is in response to the DNS request from the user's PC, let alone in response to a determination that the DNS request from the user's PC is from an authorized client.

199. Under the third embodiment of *BinGO*, I understand the Request appears to contend that the local name of the user's PC (*e.g.*, LittleIndian) and the common password between the user's PC and the WAN partner (*e.g.*, the corporate network) correspond to the plurality of authentication



tables recited in independent claim 13. (*See* Req. at 208-210.) Even assuming this were true, it is my opinion that *BinGO* nevertheless fails to disclose the features of independent claim 13.

200. For example, it is my opinion that *BinGO* fails to disclose authenticating, with reference to one of the plurality of authentication tables maintained by the central computer, that the request received in step (1) is from an authorized client, as recited in independent claim 13. I understand the Request contends that authenticating log in access to the *BinGO!* router corresponds to this feature of independent claim 13. (*Id.* at 210-211.) However, nowhere does *BinGO* disclose that authenticating log in access to the *BinGO!* router (the alleged authenticating) is related to the local name of the user's PC and the common password (the alleged plurality of authentication tables) or even the DNS request from the user's PC (the alleged request), let alone disclose authenticating log in access to the *BinGO!* router, with reference to one of the local name of the user's PC and the common password, that the DNS request is from an authorized client. The Request even fails to disclose how these three different concepts are even related. Indeed, authenticating log in access to the *BinGO!* router, the local name of the user's PC and the common password, and the DNS request from the user's PC are all concepts that are independent of one another.

201. For example, *BinGO* describes that a user may log in to the *BinGO!* router with a user name and password to configure the *BinGO!* router. (*See BinGO UG* 13-14.) Thus, the user may log in to the *BinGO!* router not by referring to the local name of the user's PC and the common password (which is used to establish a connection between the user's PC and the corporate network, for example), but simply by referring to the user's name and password. Furthermore, the user may log in to the *BinGO!* router regardless of any DNS request that is generated from the user's PC. Thus, nowhere does *BinGO* disclose that authenticating log in access to the *BinGO!* router (the alleged authenticating) is related to the local name of the user's PC and the common password (the alleged plurality of authentication tables) and/or the DNS request from the user's PC (the alleged request), let alone disclose authenticating log in access to the *BinGO!* router, with reference to one of the local name of the user's PC and the common password, that the DNS request is from an authorized client.

202. Even if the Request considers the authentication of the connection between the *BinGO!* router and router at HQ (rather than authenticating log in access to the *BinGO!* router) as corresponding to the authenticating recited in independent claim 13, it is my opinion that *BinGO* nevertheless fails to disclose the authenticating recited in independent claim 13. *BinGO* states that the router at HQ checks incoming data (*e.g.*, the local name of the user's PC and the common password) to see if it should take a call from the *BinGO!* router, and only takes the call after correct

authentication based on the local name of the user's PC and the common password). (*Id.* at 40.) While the router at HQ may authenticate the connection to the *BinGO!* router by checking the local name of the user's PC and the common password, nowhere does *BinGO* disclose that the router at HQ performs this authentication by referring to the common password that is maintained by the *BinGO!* router.

203. Furthermore, it is my opinion that *BinGO* has not been shown to disclose, responsive to a determination that the request is from an authorized client, allocating resources to establish a virtual private link between the client and a second computer, as recited in independent claim 13. Nowhere does *BinGO* disclose that the *BinGO!* router allowing a user to route traffic through it to a specified destination (the alleged allocating resources) to establish the encrypted connection between the user's PC and the BossPC (the alleged virtual private link) is in response to the DNS request from the user's PC (the alleged request), let alone in response to a determination that the DNS request from the user's PC is from an authorized client. For example, as I mentioned above with respect to independent claim 1, *BinGO* fails to disclose that an alleged VPN between the user's PC and the BossPC is automatically initiated in response to any DNS request. For at least the same reasons as independent claim 1, *BinGO* fails to disclose that the *BinGO!* router allowing a user to route traffic through it to a specified destination to establish the encrypted connection between the user's PC and the BossPC is in response to the DNS request from the user's PC, let alone in response to a determination that the DNS request from the user's PC is from an authorized client.

204. It is also my opinion that *BinGO* fails to disclose the allocating recited in independent claim 13 for additional reasons. As I mentioned above, the router at HQ checks incoming data (*e.g.*, the local name of the user's PC and the common password) to see if it should take a call from the *BinGO!* router for authentication. (*See BinGO UG 40.*) In this regard, the *BinGO!* router has at the very least transmitted the incoming data to the router at HQ. Thus, the *BinGO!* router has already allowed a user to route traffic through it to a specified destination (the alleged allocating resources) to establish the connection between the *BinGO!* router (which is connected to the user's PC) and the router at HQ (which is connected to the BossPC) before the authentication by the router at HQ.

205. Regarding dependent claim 14, I understand the Request contends that the NAT procedure described in pages 244-249 of *BinGO UG* inherently functions by changing at least one field in a series of data packets periodically according to a known sequence. (*See Req.* at 212.) I also understand the Request further points to RFC 2663 at page 1 as providing support for this contention, and therefore, the Request contends that claim 14 is anticipated by *BinGO*. (*Id.*) I

disagree.

206. I understand the Request appears to contend that an internal host address of a LAN of *BinGO* corresponds to the at least one field in a series of data packets that is periodically changed according to a known sequence, as recited in claim 14. (See Req. at 212.) However, nowhere does *BinGO* or even RFC 2663 disclose that the internal host address is periodically changed, let alone periodically changed according to a known sequence. Furthermore, while *BinGO UG* states that the internal host address of the LAN may be remapped to an external host address, *BinGO UG* also states that the external host address remains unchanged. (See *BinGO UG* 244.) Thus, *BinGO UG* describes that the internal host address is only mapped to the external address once.

207. Regarding dependent claim 15, *BinGO* describes that a user can configure the *BinGO!* router to explicitly allow a NAT interface certain IP connections to a certain internal host. (See *BinGO UG* 248.) For example, the user may specify the IP address – of the host in the LAN – in a destination field of the configuration of the *BinGO!* router. (*Id.* at 249.) If an entry is not made in the destination field, then the *BinGO!* router is assumed to be the destination. (*Id.*) I understand the Request appears to contend that the IP address of the host corresponds to the IP address in the header of each data packet as recited in claim 15, and that the *BinGO!* router corresponds to the second computer recited in claim 15. (See Req. at 212.) I also understand the Request contends that the description of the NAT procedure in *BinGO UG* at p. 249 corresponds to the feature of comparing an IP address in a header of each data packet to a table of valid IP addresses maintained in a table in the second computer. (*Id.* at 212-213.) I disagree.

208. Initially, I understand that with respect to independent claim 13, the Request considers the BossPC as corresponding to the second computer recited in independent claim 13, and the *BinGO!* router as corresponding to the central computer recited in independent claim 13. (*Id.* at 210-211.) In contrast, with respect to claim 15, which depends from claims 14 and 13, I understand the Request considers the *BinGO!* router as corresponding to the second computer, and makes no reference at all to the BossPC (*e.g.*, PC in the corporate network). (*Id.* at 212.)

209. Regarding independent claim 18, for at least similar reasons as I discussed with respect to independent claim 1, it is my opinion that *BinGO* fails to disclose a VPN between a client computer and a target computer, as recited in independent claim 18. And for at least similar reasons as I discussed with respect to independent claim 1, it is my opinion that *BinGO* has not been shown to disclose the feature of (2) determining whether the DNS request transmitted in step (1) is requesting access to a secure web site, as recited in independent claim 18. And for at least similar

reasons as I discussed with respect to independent claim 1, it is my opinion that *BinGO* fails to disclose in response to determining that the DNS request in step (2) is requesting access to a secure target web site, automatically initiating the VPN between the client computer and the target computer, as recited in independent claim 18. And for at least similar reasons as I discussed with respect to claim 5, it is my opinion that *BinGO* fails to disclose that step (3) comprises the step of, prior to automatically initiating the VPN between the client computer and the target computer, determining whether the client computer is authorized to resolve addresses of non secure target computers and, if not so authorized, returning an error from the DNS request, as recited in independent claim 18.

**F. *Reed***

**1. Claim 11**

210. The Request asserts that *Reed's* onion routing corresponds to the IP address hopping regime recited by claim 11. (*See* Req. at 110). I disagree. Nowhere does *Reed* even disclose “IP address,” let alone disclose the feature of “an IP address hopping regime that is used to pseudorandomly change IP addresses in packets transmitted between the client computer and the secure target computer,” as recited by claim 11. While *Reed* mentions “TCP/IP socket connections,” “IP Tunnel,” “IP source routing” in sections 3.1 and 8.1, these concepts are not even described by *Reed* as being related to the operation of the onion routing. Indeed, *Reed* describes that “[o]nion routing’s anonymous connections are designed to replace TCP/IP socket connections.” (*See Reed* 3.)

211. One of ordinary skill in the art can ascertain that other types of addresses besides IP addresses may be used in connection with the onion routing described in *Reed*. As discussed above, there is no mention of “IP address” in *Reed*, let alone any description in *Reed* that an IP address is necessary for the operation of the onion routing.

**a) *Aventail* in view of *Reed***

212. The Request contends that *Aventail* contemplates the use of proxy servers that route network traffic through intermediary proxy servers. (Req. at 110 (citing *Aventail v3.1* at 68).) The Request further contends that *Reed* similarly explains that its onion routing is suited to be implemented via HTTP-proxy server because *Reed* discloses that applications can “connect to onion routing’s anonymous connections using proxies.” (*Id.* (citing *Reed* ¶ 3.01).) The Request contends that one of ordinary skill in the art would have specific motivation to combine *Aventail* and *Reed* because both references identify the problem of monitoring network usage. (*Id.* at 109-10.) I disagree.

213. *Aventail* discloses that “firewalls are not designed to handle complex security issues, such as monitoring network usage [and] providing private communication over public networks . . . .” (*Aventail v3.1* 6 (emphasis added.)) *Aventail* explains that SOCKS was specifically designed to address the issues associated with monitoring network usage (*i.e.*, who can communicate with whom) and providing private communication over public networks. (*Aventail v3.1* 6-7 (“SOCKS is more than a standard firewall.” Features include authentication, encryption, UDP support, X.509 client certificates, and cross-platform support.)) *Aventail* then discloses a system which builds further on the SOCKS platform for better security. (*See, e.g., id.* at 7.) *Aventail v3.1* would not be understood to be applicable to third-parties attempting to monitor traffic flowing in or out of a firewall, or between the firewall and some remote location. (*Id.*) To the contrary, *Reed* teaches how to prevent the monitoring of network usage. *Reed* is concerned with addressing the problem of surreptitious third-parties being able to discover or “infer who is talking to whom over a public network.” (*Reed* ¶ 1.1.) Accordingly, there is no rational or objective reason why a skilled person would have looked to *Reed* to solve the same problems for which *Aventail* was designed to specifically address.

214. Additionally, *Aventail* teaches that, in a MultiProxy configuration, in order to traverse multiple firewalls, “*Aventail* Connect makes a connection with each proxy server individually,” so that “each proxy server in a chain can provide authentication, access control, and encryption” to the client computer. (*Aventail v3.1* 68, 72.) This is because, the client computer must authenticate with each proxy server using its own proprietary authentication scheme. (*See, e.g., id.* at 13, 46-73 (describing authentication modules, encryption, certificates, server validation, credential timeouts.)) To the contrary, *Reed* teaches that onion routing provides anonymous connections; *i.e.*, “[i]nstead of containing source and destination information, packets moving along an anonymous connection contain only next hop and previous hop information.” (*See, e.g., Reed* ¶ 1.2.) Consequently, the onion routing infrastructure of *Reed*, which prevents a router from learning about other routers in a chain, is directly opposed to the teaching of *Aventail* which specifically teaches that the client must authenticate with each server. (*See Aventail v3.1* 73 (“Client must be aware of Server 1 and Server 2.”))

215. Moreover, the proxy chaining feature of *Aventail* is disclosed as being configured to allow traffic between two extranets. (*See Aventail v3.1* 71-72.) Even if an *Aventail* Extranet Server could be configured as an onion router, as the Office Action suggests, any modification of *Aventail* to include multiple “extranet servers” (onion routers) in a communication path between two extranets

would drastically change the function of the subject matter of *Aventail* and make each extranet server that functions as a router unsatisfactory for its intended purpose of providing secure access to a private extranet. (*See id.* at 77.) Moreover, a proposed modification of *Aventail* to employ Aventail ExtraNet Servers merely to provide the anonymous routing disclosed by *Reed* would not only be an illogical reading of *Aventail* and *Reed*, but cost prohibitive.

216. With regard to claim 14, the Request contends that the Aventail ExtraNet Server (characterized by the Request as a “VPN server”) may function as a central computer to establish VPNs between client computers and secure destination computers. (Req. at 113.) I disagree.

217. In my opinion, the Requests shifts its previous arguments to reflect that a second computer is no longer seen as a destination computer or remote host, but, rather, now the second computer may be an “intermediary destination” along a routing path. (*Compare* Req. IV.A-C. with Req. IV.D.)

218. As I previously explained, *Aventail* does not disclose at least the step of “communicating according to a scheme by which at least one field in a series of data packets is periodically changed according to a known sequence,” *Reed* discloses an infrastructure called *onion routing* by which an initiator computer chooses a route through multiple routers and creates a layered data structure called an onion that designates each router in the chain. (*E.g.*, *Reed* ¶ 4.2, 5.1.) *Reed* teaches that *onion routing* provides anonymous connections; *i.e.*, “[i]nstead of containing source and destination information, packets moving along an anonymous connection contain only next hop and previous hop information.” (*See, e.g.*, *Reed* ¶ 1.2.) Thus, each hop (router) is prevented from learning about other routers in the chain.

219. Even if *Aventail* could be construed to establish a virtual private link between the client and a second computer, that link would not be to some intermediary device. (*See, e.g.*, *Aventail* v3.1 12 (“The application requests a connection to the remote host.”).) Moreover, the Request does not describe how the central computer may receive a request to establish a connection from one of the plurality of client computers, the request be authenticated, and then resources allocated to establish a virtual private link between the client and a second computer, now contended to be an intermediary device between the client and the central computer (Aventail ExtraNet Server). The proposed configuration would be a technical impossibility, as *Reed* specifically teaches that only a client computer who starts the connection knows about other routers in a chain. A central computer could not receive a request to establish a connection, especially if the onion restricts all devices except for the client, including a central computer, from learning about any other intermediary

devices in the chain.

b) **BinGO in view of Reed**

220. I understand the Request concedes that *BinGO* fails to disclose the feature in which “the gatekeeper computer creates the VPN by establishing an IP address hopping regime that is used to pseudorandomly change IP addresses in packets transmitted between the client computer and the secure target computer,” as recited in claim 11. (*See* Req. at 218). I also understand the Request contends that the onion routing scheme of *Reed* may be applied to the *BinGO!* router of *BinGO*. (*Id.*) This proposed modification, however, would render the *BinGO!* router unsatisfactory for its intended purpose.

221. As I mentioned above, the *BinGO!* router connects with either the router of an ISP or the router of a corporate network via ISDN. (*See Bingo UG* 15-16). For example, *BinGO* describes setting up the *BinGO!* router by having the user enter the dial number of the ISP or the corporate network into a configuration wizard. (*Id.* at 39-40) Thus, the *BinGO!* router establishes the connection to the ISP or the corporate network by dialing directly in to the desired destination. (*Id.* at 17.) However, the onion routing of *Reed* is described to provide “anonymous connections.” (*See Reed* 2.) *Reed* also describes that under the onion routing, “[i]nstead of containing source and destination information, packets moving along an anonymous connection contain only next hop and previous hop information.” (*Id.* at 2.) Thus, if the onion routing scheme of *Reed* were to be combined with the *BinGO!* router of *BinGO*, the *BinGO!* router would not be able to directly dial in to the desired destination, thereby rendering the *BinGO!* router unsatisfactory for its intended purpose to establish a connection to the ISP or the corporate network. Thus, there can be no suggestion or motivation to apply the onion routing scheme of *Reed* to the *BinGO!* router.

c) **Reed in further view of Goldschlag**

222. For the following reasons, I disagree that the Request establishes disclosure of the gatekeeper *computer* creates the VPN by establishing an IP address *hopping* regime that *is* used to pseudorandomly change IP addresses in packets transmitted between the client computer and the secure target computer, as recited by claim 11.

223. The Request relies on *Goldschlag* to support the contention that *Reed* discloses creating a pseudorandom *path* by allowing nodes to “choose their own route” and *add* more hops to the chain.” (Req. at 104-05.) *Goldschlag* merely hypothesizes that nodes may be instructed to “choose their own route.” (*Goldschlag* at 6-7.) *Goldschlag* does not disclose how an IP address in a packet could be pseudorandomly changed. The Request also does not specify how pseudorandomly

changing *IP* addresses *in packets* would be necessarily present in *Goldschlag*, especially when the route chosen by the node may be predefined by the node.

**2. Claim 15**

224. With regard to claim 15, the Request asserts that *Reed* discloses onion routing schemes that “will compare IP address to tables of IP addresses maintained by intermediate onion routers. . . .” (Req. at 113.) In my opinion, this does not show comparing an Internet Protocol (IP) address in a header of each data packet to a table of valid IP addresses maintained in a table in the second computer, as recited by claim 15.

225. *Reed* teaches that data travels in a series of fixed size cells, and that cells from one anonymous connection may be transmitted over the same socket connection as cells from another anonymous connection. (*Reed* ¶ 4.1.) Each cell contains an identifier of the anonymous connection that the cell is assigned to so that cells may be forwarded to neighboring onion routers without losing their relationship to each other. (*Reed* ¶ 5.2.1.) In this respect, each onion router maintains a table that maps between the identifiers of incoming connections and outgoing connections. (*Id.*) Thus, where a single router may forward data from multiple connections, each connection will be associated with a different identifier. (*Id.*) When a data cell arrives at a neighboring onion router, the onion router looks up the cell’s identifier in its tables and finds the corresponding outbound identifier. (*Reed* ¶ 5.3.1.)

226. The argument that *Reed* teaches the subject matter of claim 15 fails at least because the identifier of *Reed* is not an IP address. The identifier merely informs the onion router what cells are associated with each other so that a router may distinguish between multiplexed connections. (*Reed* ¶ 5.2.1.)

**G. *Boden* (claim 16)**

227. I disagree that the Request establishes disclosure of comparing the IP address in the header of each data packet to a moving window of valid IP addresses, and rejecting data packets having IP addresses that do not fall within the moving window, as recited by claim 16.

228. The Request asserts that *Boden* teaches methods for improving security by employing a process in which IP address information the header of data packets are compared to a dynamically changing pool (a ‘window’) of valid IP addresses. I disagree.

229. *Boden* is generally directed to dynamically generating NAT rules and associating them with the manual or dynamically generated (IKE) Security Associations. (*Boden* 2:54-56.) The relevant parts of *Boden* teach that a user defines a pool, or range, of IP addresses that may be



associated with remote ID and local ID database entities. (*Id.* at 4:51-56.) Boden explains:

When starting an initiator mode connection, the connection manager checks if the local client ID is to be translated. If so, the connection manager looks for an available IP address from NAT pool . . . associated with a remote ID in the database. [T]he connection manager . . . maintains a . . . list of IP addresses that have been used in some active connection . . . . The first IP address in the pool not in the used list, is chosen, and added to the used list. If an available IP address cannot be found, the connection is not started and an appropriate error message . . . is generated.

Accordingly, when a connection is started an IP address is retrieved from a pool associated with the destination. If no addresses are available then an error is generated.

230. *Boden* does not compare an IP address, and certainly does not compare an IP address in a header of a data packet to a moving window. IP addresses are not compared, but, rather, retrieved based on an ID. Retrieving an IP address from a fixed pool is not the same as comparing an IP address to a moving window, and certainly not an IP address in a header of a data packet. *Boden* does not even disclose an IP address in a data packet, and certainly not comparing an IP address. Moreover, *Boden* does not disclose a moving window of IP addresses. Even if a pool of IP addresses could be seen as a window, the Request does not show that that the pool is a “moving” window.

**H. *Weiss* (claim 17)**

231. *Weiss* is directed to an apparatus and method for the electronic generation of variable, non-predictable codes and the validation and comparison of such codes for the purpose of positively identifying an authorized individual or use of an apparatus or system. (*Weiss* 1:15-19.) In particular, *Weiss* describes a system for comparing and matching non-predictable codes generated by separate computers on the basis of dynamic variables defined by separate clock mechanisms according to time. (*Id.* at 2:3-7.) *Weiss* explains:

The present invention eliminates the relatively easy access afforded to someone who copies or otherwise misappropriates a secret “fixed” code by periodically generating identification codes by using fixed codes, variable data, and a predetermined algorithm which is unknown in advance and unknowable outside the administration of the security system even to authorized users of the apparatus utilizing the fixed secret code. The predetermined algorithm constantly generates new unique and verifiable non-predictable codes, which are derived from the fixed data and at least one dynamic variable, such as the time of day (including the date) by the predetermined algorithm.

(*Id.* at 1:55-67.)

232. The Request asserts that the non-predictable codes correspond to the periodically changing parameter recited by claim 17, and that the token authentication processes described in

*Weiss* correspond to the feature of using a checkpoint data structure that maintains synchronization of a periodically changing parameter known by the central computer and the client computer to authenticate the client, as recited by claim 17. (See Req. at 221-22.) I disagree.

233. As discussed above, *Weiss* describes that the predetermined algorithm constantly generates new unique and verifiable non-predictable codes, which are derived from the fixed data and at least one dynamic variable. Thus, these non-predictable codes do not change, let alone periodically change. Rather, each of these non-predictable codes is a new and unique non-predictable code that is generated each time the predetermined algorithm processes the fixed data and the dynamic variable.

234. For example, *Weiss* describes that an authorized person may be provided with a fixed secret code or card seed 10, which is unique to that individual. (*Id.* at 5:8-11.) This user may input the card seed 10, together with a personal identification number (PIN) 45, into a credit card sized computer 20 and a host computer 50 (also referred to as an access control module) in order to generate a non-predictable code 40, which will ultimately give the user clearance or access to an authorized terminal. (*Id.* at 5:27-59, Figures 1, 1A, and 2.) Each time the card seed 10 and the PIN 45 are inputted as static variables, a predetermined algorithm on the credit card sized computer 20 and the host computer 50 utilizes a second dynamic variable (*e.g.*, a number defined and determined by the interval of time in which the card seed 10 and/or PIN 45 is inputted) to generate the non-predictable code 40. (*Id.* at 6:28-66, Figures 1, 1A, and 2.) Thus, the non-predictable codes of *Weiss* do not change. Rather, each of the non-predictable codes is newly generated when the predetermined algorithm processes (1) the static variables that are inputted and (2) the dynamic variable that is utilized at the time the static variables are inputted.

235. Even if the generation of new and unique non-predictable codes can be considered as changing, the non-predictable codes do not change periodically. As discussed above, each non-predictable code of *Weiss* is generated each time the predetermined algorithm processes (1) the static variables that are inputted and (2) the dynamic variable that is utilized at the time the static variables are inputted. Nowhere, however, does *Weiss* disclose that the static variables are inputted periodically so that the predetermined algorithm may generate a new non-predictable code periodically. Indeed, it is the user who determines whether or not to input the static variables, and nowhere does *Weiss* disclose that the user inputs the static variables periodically. Accordingly, the non-predictable codes of *Weiss* do not periodically change.

236. Furthermore, the non-predictable codes of *Weiss* are not known by the computers of

*Weiss*. Indeed, these codes are described as being non-predictable. Thus, the credit card sized computer 20 and the host computer 50, for example, will not know in advance what a particular non-predictable code will be. As discussed above, the predetermined algorithm on the credit card sized computer 20 and the host computer 50 only generate the non-predictable code when the static variables and the dynamic variable are inputted. Thus, none of the computers (or even the user) know what the non-predictable code will be. *Weiss* even states that the predetermined algorithm is “unknown in advance and unknowable outside the administration of the security system even to authorized users of the apparatus utilizing the fixed secret code.” (*Id.* at 1:59-62.) *Weiss* describes that making the non-predictable code unknown eliminates the relatively easy access afforded to someone who copies or otherwise misappropriates a secret fixed code. (*Id.* at 1:55-57.)

#### **Truth and Accuracy of Statements**

I further 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 willful false statements or the like may jeopardize the validity of the ‘135 patent.

Signed at New York, New York, this 15th day of May, 2012.

/Angelos D. Keromytis/  
Angelos D. Keromytis

## Angelos D. Keromytis - *Curriculum Vitae*

### Positions Held

- **January 2006 - Present**  
Associate Professor, Department of Computer Science, Columbia University, New York.
- **January 2009 - January 2010**  
Senior Research Engineer, Symantec Research Labs Europe, Sophia Antipolis, France.
- **July 2001 - December 2005**  
Assistant Professor, Department of Computer Science, Columbia University, New York.
- **September 1996 - July 2001**  
Research Assistant, Computer and Information Science Department, University of Pennsylvania, Philadelphia.
- **January 1993 - October 1995**  
Member of the Technical Staff, FORTHnet S.A., Heraclion, Greece.
- **September 1991 - January 1993**  
Member of the Technical Staff, Education Team, Computer Center of the University of Crete, Heraclion, Greece.

### Education

- **November 2001**  
Ph.D. (Computer Science), University of Pennsylvania, USA.
- **August 1997**  
M.Sc. (Computer Science), University of Pennsylvania, USA.
- **June 1996**  
B.Sc. (Computer Science), University of Crete, Greece.

### Service and Teaching

#### Editorial Boards and Steering Committees

- Associate Editor, *Encyclopedia of Cryptography and Security* (2<sup>nd</sup> Edition), Springer, 2010 - 2011.
- Associate Editor, IET (formerly IEE) *Proceedings Information Security*, 2005 - 2010.
- Steering Committee, *ISOC Symposium on Network and Distributed System Security (SNDSS)*, 2006 - 2009.
- Steering Committee, *New Security Paradigms Workshop (NSPW)*, 2007 onward.
- Associate Editor, *ACM Transactions on Information and System Security (TISSEC)*, 2004 - 2010.
- Steering Committee, *USENIX Workshop on Hot Topics in Security (HotSec)*, 2006 - 2009.
- Steering Committee, *Computer Security Architecture Workshop (CSAW)*, 2007 - 2009.

#### Program Chair

- Program Chair, 16<sup>th</sup> International Conference on Financial Cryptography and Data Security (FC), 2012.
- Program co-Chair, 17<sup>th</sup> ACM Computer and Communication Security (CCS), 2010.
- Program co-Chair, 16<sup>th</sup> ACM Computer and Communication Security (CCS), 2009.

- Program co-Chair, New Security Paradigms Workshop (NSPW), 2008.
- Program co-Chair, New Security Paradigms Workshop (NSPW), 2007.
- Chair, 27<sup>th</sup> International Conference on Distributed Computing Systems (ICDCS), *Security Track*, 2007.
- Chair, 16<sup>th</sup> World Wide Web (WWW) Conference, *Security, Privacy, Reliability and Ethics Track*, 2007.
- Chair, 15<sup>th</sup> USENIX Security Symposium, 2006.
- Deputy Chair, 15<sup>th</sup> World Wide Web (WWW) Conference, *Security, Privacy and Ethics Track*, 2006.
- Chair, 3<sup>rd</sup> Workshop on Rapid Malcode (WORM), 2005.
- Program co-Chair, 3<sup>rd</sup> Applied Cryptography and Network Security (ACNS) Conference, 2005.
- Program co-Chair, OpenSig Workshop, 2003.

### **Program Organization**

- General Chair, New Security Paradigms Workshop (NSPW), 2010.
- General Vice Chair, New Security Paradigms Workshop (NSPW), 2009.
- Co-chair, Invited Talks, 17<sup>th</sup> USENIX Security Symposium, 2008.
- General co-chair, Applied Cryptography and Network Security (ACNS) Conference, 2008.
- Co-chair, Invited Talks, 16<sup>th</sup> USENIX Security Symposium, 2007.
- Organizing Committee, Columbia/IBM/Stevens Security & Privacy Day (bi-annual event).
  - Organizer, Columbia/IBM/Stevens Security & Privacy Day, December 2010.
  - Organizer, Columbia/IBM/Stevens Security & Privacy Day, June 2007.
- Co-organizer, ARO/FSTC Workshop on Insider Attack and Cyber Security, 2007.
- Publicity co-Chair, ACM Conference on Computer and Communications Security, 2006.
- General co-Chair, OpenSig Workshop, 2003.

### **Program Committees**

- Program Committee, ISOC Symposium on Network and Distributed Systems Security (SNDSS), 2003, 2004, 2006, 2007, 2008, 2012.
- Program Committee, International Workshop on Security (IWSEC), 2006, 2007, 2008, 2009, 2010, 2011.
- Program Committee, ACM Conference on Computer and Communications Security (CCS), 2005, 2007, 2008, 2009, 2010.
- Program Committee, Applied Cryptography and Network Security (ACNS) Conference, 2005, 2006, 2010, 2011, 2012.
- Program Committee, USENIX Security Symposium, 2004, 2005, 2006, 2008.
- Program Committee, International Conference on Distributed Computing Systems (ICDCS), *Security Track*, 2005, 2006, 2007, 2008.
- Program Committee, Workshop on Rapid Malcode (WORM), 2004, 2005, 2006, 2007.
- Program Committee, Information Security Conference (ISC), 2005, 2007, 2009, 2011.
- Program Committee, World Wide Web Conference (WWW), 2005, 2006, 2007.
- Program Committee, USENIX Workshop on Hot Topics in Security (HotSec), 2006, 2007, 2010.
- Program Committee, Financial Cryptography (FC) Conference, 2002, 2010, 2011, 2012.
- Program Committee, European Workshop on Systems Security (EuroSec), 2009, 2010, 2011.

- Program Committee, Annual Computer Security Applications Conference (ACSAC), 2006, 2007, 2011.
- Program Committee, USENIX Technical Conference, *Freely Distributable Software (Freenix) Track*, 1998, 1999, 2003.
- Program Committee, IEEE Security & Privacy Symposium, 2006, 2008.
- Program Committee, ACM SIGCOMM Workshop on Large Scale Attack Defense (LSAD), 2006, 2007.
- Program Committee, New Security Paradigms Workshop (NSPW), 2007, 2008.
- Program Committee, IEEE WETICE Workshop on Enterprise Security, 2002, 2003.
- Program Committee, International Conference on Mathematical Methods, Models and Architectures for Computer Network Security (MMM-ACNS), 2007, 2010.
- Program Committee, USENIX Annual Technical Conference (ATC), 2008, 2011.
- Program Committee, European Symposium on Research in Computer Security (ESORICS), 2011.
- Program Committee, International Workshop on Mobile Security (WMS), 2010.
- Program Committee, 40<sup>th</sup> Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), Dependable Computing and Communication Symposium (DCCS), 2010.
- Program Committee, Computer Forensics in Software Engineering Workshop, 2009.
- Program Committee, USENIX Workshop on Large-scale Exploits and Emergent Threats (LEET), 2008.
- Program Committee, 23<sup>rd</sup> International Information Security Conference (IFIP SEC), 2008.
- Program Committee, Joint iTrust and PST Conferences on Privacy, Trust Management and Security (IFIPTM), 2008.
- Program Committee, 1<sup>st</sup> Computer Security Architecture Workshop (CSAW), 2007.
- Program Committee, 8<sup>th</sup> IEEE Information Assurance Workshop (IAW), 2007.
- Program Committee, Anti-Phishing Working Group (APWG) eCrime Researchers Summit, 2007.
- Program Committee, 4<sup>th</sup> GI International Conference on Detection of Intrusions & Malware, and Vulnerability Assessment (DIMVA), 2007.
- Program Committee, 2<sup>nd</sup> ACM Symposium on Information, Computer and Communications Security (AsiaCCS), 2007.
- Program Committee, 6<sup>th</sup> International Conference on Cryptology and Network Security (CANS), 2007.
- Program Committee, 2<sup>nd</sup> Workshop on Advances in Trusted Computing (WATC), 2006.
- Program Committee, International Conference on Information and Communications Security (ICICS), 2006.
- Program Committee, 2<sup>nd</sup> Workshop on Secure Network Protocols (NPsec), 2006.
- Program Committee, 1<sup>st</sup> Workshop on Hot Topics in System Dependability (HotDep), 2005.
- Program Committee, 20<sup>th</sup> ACM Symposium on Applied Computing (SAC), Trust, Recommendations, Evidence and other Collaboration Know-how (TRECK) Track, 2005.
- Program Committee, 1<sup>st</sup> Workshop on Operating System and Architecture Support for the on demand IT Infrastructure (OASIS), 2004.
- Program Committee, Workshop on Information Security Applications (WISA), 2004.
- Program Committee, Workshop on Logical Foundations of an Adaptive Security Infrastructure (WOLFASI), 2004.
- Program Committee, 29<sup>th</sup> IEEE Conference on Local Computer Networks (LCN), 2004.
- Program Committee, 2<sup>nd</sup> International Conference on Trust Management, 2004.

- Program Committee, Asia BSD Conference, 2004.
- Program Committee, 2<sup>nd</sup> Annual New York Metro Area Networking Workshop (NYMAN), 2002.
- Program Committee, Cloud Computing Security Workshop (CCSW), 2009.
- Program Committee, Workshop on Grid and Cloud Security (WGC-Sec), 2011.
- Program Committee, Workshop on Cyber Security Experimentation and Test (CSET), 2011.

### **Advisory Workshops**

- ODNI/NSA Invitational Workshop on Computational Cybersecurity in Compromised Environments (C3E), Keystone, CO, September 2011.
- ONR Workshop on Host Computer Security, Chicago, IL, October 2010.
- Intel Workshop on Trust Evidence and End-to-end Trust in Heterogeneous Environments, Santa Clara, CA, May 2010.
- Intelligence Community Technical Exchange on Moving Target, Washington, DC, April 2010.
- Lockheed Martin Future Security Threats Workshop, New York, NY, November 2009.
- Air Force Office for Scientific Research (AFOSR) Invitational Workshop on Homogeneous Enclave Software vs Heterogeneous Enclave Software, Arlington, VA, October 2007.
- NSF Future Internet Network Design Working Meeting, Arlington, VA, June 2007.
- ARO/FSTC Workshop on Insider Attack and Cyber Security, Arlington, VA, June 2007.
- NSF Invitational Workshop on Future Directions for the CyberTrust Program, Pittsburgh, PA, October 2006.
- ARO/HSARPA Invitational Workshop on Malware Detection, Arlington, VA, August 2005.
- Department of Defense Invitational Workshop on the Complex Behavior of Adaptive, Network-Centric Systems, College Park, MD, July 2005.
- ARDA Next Generation Malware Invitational Workshop, Annapolis Junction, MD, March 2005.
- Co-leader of session on "Securing software environments", joint NSF and Department of Treasury Invitational Workshop on Resilient Financial Information Systems, Washington, DC, March 2005.
- DARPA Application Communities Invitational Workshop, Arlington, VA, October 2004.
- DARPA APNets Invitational Workshop, Philadelphia, PA, December 2003.
- NSF/NIST Invitational Workshop on Cybersecurity Workforce Needs Assessment and Educational Innovation, Arlington, VA, August 2003.
- NSF Invitational Workshop on Large Scale Cyber-Security, Lansdowne, VA, March 2003.
- IP Security Working Group Secretary, Internet Engineering Task Force (IETF), 2003 - 2008.
- Session moderator, Workshop on Intelligence and Research, Florham Park, NJ, October 2001.
- DARPA Composable High Assurance Trusted Systems #2 (CHATS2) Invitational Workshop, Napa, CA, November 2000.

### **Other Professional Activities**

- Co-chair, ACM Computing Classification System Update Committee ("Security and Privacy" top-level node), 2011.
- Member, ACM Computing Classification System Update Committee (top two levels), 2010.
- External Advisory Board member, *"i-code: Real-time Malicious Code Identification"*, EU

- project, 2010 - 2012.
- Reviewer (grant applications), Greek Ministry of Education, 2010.
- Reviewer (grant applications), Danish National Research Foundation, 2010.
- Member of the Scientific Advisory Board, Centre for Research and Technology, Hellas (CERTH), 2008 - 2011.
- Senior Member of the ACM, 2008 onward.
- Senior Member of the IEEE, 2009 onward.
- Visiting Scientist, Institute for Infocomm Research (I<sup>2</sup>R), Singapore, February - May 2007.
- Columbia Representative to the Institute for Information Infrastructure Protection (I3P), 2006 - 2008.
- Technical Advisory Board, *StackSafe Inc. (formerly Revive Systems Inc.)*, 2006 - 2009.
- Technical Advisory Board, *Radiuz Inc.*, 2006.
- Reviewer (grant applications), Institute for Security Technology Studies (ISTS), Dartmouth College, 2006.
- Reviewer, Singapore National Science and Technology Awards (NSTA), 2006.
- Board of Directors, *StackSafe Inc. (formerly Revive Systems Inc.)*, 2005 - 2009.
- Founder, *StackSafe Inc. (formerly Revive Systems Inc.)*, 2005 - 2009.
- Expert witness in criminal and intellectual property litigation cases, 2005, 2006, 2007, 2009, 2010, 2011.
- Science Fair Judge, Middle School for Democracy and Leadership, Brooklyn, NY, 2005, 2006.
- Reviewer (grant applications), Swiss National Science Foundation, 2007.
- Reviewer (grant applications), Netherlands Organisation for Scientific Research, 2005, 2006.
- Reviewer (grant applications), US/Israel Binational Science Foundation, 2003, 2005.
- NSF reviewer & panelist, 2002, 2003, 2006, 2008, 2009, 2011.
- Internet Engineering Task Force (IETF) Security Area Advisor, 2001 - 2008.

#### **Ph.D. Thesis Committee Service**

- Michalis Polychronakis, "*Generic Code Injection Attack Detection using Code Emulation*", Computer Science Department, University of Crete, October 2009.
- Spyros Antonatos, "*Defending against Known and Unknown Attacks using a Network of Affined Honey pots*", Computer Science Department, University of Crete, October 2009.
- Van-Hau Pham, "*Honey pot Traces Forensics by Means of Attack Event Identification*", Computer Science Group, Communications and Electronics Department, Ecole Nationale Supérieure des Telecommunications, September 2009.
- Gabriela F. Ciocarlie, "*Towards Self-Adaptive Anomaly Detection Sensors*", Department of Computer Science, Columbia University, September 2009.
- Vanessa Frias-Martinez, "*Behavior-Based Admission and Access Control for Network Security*", Department of Computer Science, Columbia University, September 2008.
- Wei-Jen Li, "*SPARSE: A Hybrid System for Malcode-Bearing Document Detection*", Department of Computer Science, Columbia University, June 2008.
- Raj Kumar Rajendran, "*The Method for Strong Detection for Distributed Routing*", Electrical Engineering Department, Columbia University, March 2008.
- Constantin Serban, "*Advances in Decentralized and Stateful Access Control*", Computer Science Department, Rutgers University, December 2007.
- Ricardo A. Baratto, "*THINC: A Virtual and Remote Display Architecture for Desktop Computing*", Computer Science Department, Columbia University, October 2007.



- Zhenkai Liang, "*Techniques in Automated Cyber-Attack Response and Recovery*", Computer Science Department, Stony Brook University, November 2006.
- Ke Wang, "*Network Payload-based Anomaly Detection and Content-based Alert Correlation*", Computer Science Department, Columbia University, August 2006.
- Seoung-Bum Lee, "*Adaptive Quality of Service for Wireless Ad hoc Networks*", Electrical Engineering Department, Columbia University, June 2006.
- Shlomo Hershkop, "*Behavior-based Email Analysis with Application to Spam Detection*", Computer Science Department, Columbia University, August 2005.
- Gaurav S. Kc, "*Defending Software Against Process-subversion Attacks*", Computer Science Department, Columbia University, April 2005.
- Gong Su, "*MOVE: A New Virtualization Approach to Mobile Communication*", Computer Science Department, Columbia University, May 2004.
- Jonathan M. Lennox, "*Services for Internet Telephony*", Computer Science Department, Columbia University, December 2003.
- Michael E. Kounavis, "*Programming Network Architectures*", Electrical Engineering Department, Columbia University, June 2003.
- Wenyu Jiang, "*QoS Measurement and Management for Internet Real-time Multimedia Services*", Computer Science Department, Columbia University, April 2003.

#### **Post-doctoral Students**

- Hyung Chan Kim (October 2007 - October 2008)
- Stelios Sidiroglou (October 2008 - December 2008)
- Georgios Portokalidis (March 2010 - present)
- Michalis Polychronakis (May 2010 - present)
- Dimitris Geneiatakis (June 2010 - present)

#### **Current Ph.D. Students**

- Georgios Kontaxis (September 2011)
- Vasilis Pappas (September 2009 - present)
- Vasileios Kemerlis (September 2008 - present)
- Kangkook Jee (January 2008 - present)
- Sambuddho Chakravarty (January 2007 - present)
- Angelika Zavou (September 2006 - present)

#### **Graduated Ph.D. Students**

- Debra Cook (January 2002 - June 2006)
  - Thesis title: "*Elastic Block Ciphers*"
  - Post-graduation: Member of the Technical Staff, Bell Labs
  - Currently: Research Staff Member, Telcordia Research
- Angelos Stavrou (January 2003 - August 2007)
  - Thesis title: "*An Overlay Architecture for End-to-End Service Availability*" (awarded with distinction)
  - Post-graduation: Assistant Professor, Computer Science Department, George Mason University (GMU)

- Currently: Assistant Professor, Computer Science Department, George Mason University (GMU)
- Michael E. Locasto (September 2002 - December 2007)
  - Thesis title: *"Integrity Postures for Software Self-Defense"* (awarded with distinction)
  - Post-graduation: ISTS Research Fellow, Dartmouth College
  - Currently: Assistant Professor, Department of Computer Science, University of Calgary
- Stelios Sidiroglou (June 2003 - May 2008)
  - Thesis title: *"Software Self-healing Using Error Virtualization"*
  - Post-graduation: Research Scientist, Columbia University
  - Currently: Research Scientist, MIT CSAIL
- Mansoor Alicherry (September 2006 - October 2010)
  - Thesis title: *"A Distributed Policy Enforcement Architecture for Mobile Ad Hoc Networks"*
  - Post-graduation: Member of the Technical Staff, Alcatel-Lucent Bell Labs
  - Currently: Member of the Technical Staff, Alcatel-Lucent Bell Labs
- Brian Bowen (September 2007 - December 2010; co-advised with Salvatore J. Stolfo)
  - Thesis title: *"Design and Analysis of Decoy Systems for Computer Security"*
  - Post-graduation: Member of the Technical Staff, Sandia National Laboratories
  - Currently: Member of the Technical Staff, Sandia National Laboratories

### Service at Columbia

- Computer Science Department Ph.D. Committee, 2010 - 2011
- Computer Science Department Facilities committee, 2001 - 2008, 2010 - current
  - Chair, Facilities committee, 2003 - 2005, 2011 - current
- M.Sc. Admissions committee, 2007 - current.
- M.Sc. Committee, 2008 - current.
- Computer Science Department Faculty Recruiting committee, 2002, 2008
- Columbia committee on Research Conflict of Interest Policy, 2007 - 2008
- Co-organizer, Computer Science Faculty Retreat, Fall 2007
- Advisor for the School of Engineering Computer Science Majors, Freshmen & Sophomores, 2004 - 2005
- Computer Science Department Undergraduate Admissions Representative, 2003 - 2008
- Advisor for the School of Engineering Computer Science Majors, Seniors, 2003 - 2004, 2006 - 2007
- Computer Science Department Space Allocation Policy committee, 2002 - 2010
- Computer Science Department Events Representative, 2002 - 2008
- Advisor for the School of Engineering Computer Science Majors, Juniors, 2002 - 2003, 2005 - 2006
- Computer Science Department CRF Director Hiring committee, 2003
- Advisor for the School of Engineering Computer Science Majors, Sophomores, 2001 - 2002
- Computer Science Department Faculty Recruiting committee, 2001 - 2002
- Executive Vice Provost committee on Columbia's response to the 9/11 events, Fall 2001

### Teaching

*(Scores indicate mean course quality rating from student survey; survey not conducted for summer*

sessions)

- Instructor, COMS E6183-1 - Advanced Topics in Network Security, Columbia University
  - Fall 2006: 17 on-campus students (4.58/5)
- Instructor, COMS W6998.1 - Advanced Topics in Network Security, Columbia University
  - Fall 2004: 17 on-campus students (4.62/5)
  - Spring 2003: 18 on-campus students (N/A)
- Instructor, COMS W4180 - Network Security, Columbia University
  - Spring 2011: 4 CVN students (N/A)
  - Fall 2010: 2 CVN students (N/A)
  - Spring 2010: 25 on-campus and 5 CVN students (4.48/5)
  - Summer 2006: 7 CVN students (N/A)
  - Spring 2006: 63 on-campus and 9 CVN students (4.14/5)
  - Summer 2005: 4 CVN students (N/A)
  - Spring 2005: 41 on-campus and 5 CVN students (4.25/5)
  - Summer 2004: 6 CVN students (N/A)
  - Fall 2003: 45 on-campus and 12 CVN students (3.74/5)
  - Summer 2003: 5 CVN students (N/A)
  - Fall 2002: 43 on-campus and 9 CVN students (3.21/5)
  - Fall 2001: 23 on-campus students (3.6/5)
- Instructor, COMS W4118 - Operating Systems, Columbia University
  - Summer 2007: 8 CVN students (N/A)
  - Fall 2006: 59 on-campus and 7 CVN students (3.73/5)
  - Summer 2006: 15 CVN students (N/A)
  - Fall 2005: 52 on-campus and 9 CVN students (3.86/5)
  - Spring 2004: 32 on-campus and 4 CVN students (3.39/5)
  - Spring 2002: 37 on-campus students (3.13/5)
- Instructor, COMS W3157 - Advanced Programming, Columbia University
  - Fall 2010: 37 on-campus students (3.25/5)
  - Fall 2007: 30 on-campus students (4.16/5)
- Instructor, CIS700/002 - Building Secure Systems, University of Pennsylvania, Spring 1998

#### Support for Research and Teaching (Gifts and Grants)

1. PI (co-PIs: Roxana Geambasu, Junfeng Yang, Simha Sethumadhavan, Sal Stolfo), *"MEERKATS: Maintaining EnterprisE Resiliency via Kaleidoscopic Adaptation & Transformation of Software Services"*, DARPA MRC, **\$6,619,270** (09/2011 - 09/2015; leading team that includes George Mason University and Symantec Corp.)
2. PI, *"NSF Support for the 2011 New Security Paradigms Workshop Financial Aid (Supplement)"*, NSF Trustworthy Computing, **\$10,000** (06/2011 - 07/2012)
3. PI, *"Leveraging the Cloud to Audit Use of Sensitive Infomation"*, Google (research gift), **\$60,200** (05/2011)
4. co-PI (with Sal Stolfo), *"ADAMS Advanced Behavioral Sensors (ABS)"*, DARPA ADAMS, **\$780,996** (05/2011 - 04/2013)
5. PI, *"Tracking Sensitive Information Flows in Modern Enterprises"*, Intel, **\$84,951** (12/2010 - 12/2011)
6. co-PI (with Simha Sethumadhavan, Sal Stolfo, Junfeng Yang, and David August @ Princeton), *"SPARCHS: Symbiotic, Polymorphic, Autotomic, Resilient, Clean-slate, Host*

- Security*", DARPA CRASH, **\$6,424,180** (10/2010 - 09/2014)
7. PI, "*NSF Support for the 2010 New Security Paradigms Workshop Financial Aid*", NSF Trustworthy Computing, **\$10,000** (09/2010 - 08/2011)
  8. PI (co-PIs: Junfeng Yang, Sal Stolfo), "*MINESTRONE*", IARPA, **\$7,530,113** (08/2010 - 07/2014; leading team that includes Stanford University, George Mason University, and Symantec Corp.)
  9. co-PI (with Junfeng Yang and Dawson Engler @ Stanford), "*Seed: CSR: Large: Collaborative Research: SemGrep: Improving Software Reliability Through Semantic Similarity Bug Search*", NSF CSR, CNS-10-12107, **\$325,000** (07/2010 - 06/2011)
  10. PI, "*Tracking Sensitive Information Flows in Modern Enterprises*", Intel, **\$82,286** (08/2009 - 07/2010)
  11. PI, "*Supplement for International Research Collaborations*", NSF Trustworthy Computing, **\$41,769** (09/2009 - 08/2011)
  12. PI, "*NSF Support for the 2009 New Security Paradigms Workshop Financial Aid*", NSF Trustworthy Computing, **\$10,000** (09/2009 - 08/2010)
  13. PI, "*Measuring the Health of Internet Routing: A Longitudinal Study*", Google (research gift), **\$60,000** (07/2009)
  14. PI, "*CSR: Small: An Information Accountability Architecture for Distributed Enterprise Systems*", NSF Trustworthy Computing, CNS-09-14312, **\$450,000** (07/2009 - 06/2012)
  15. co-PI (with Jason Nieh), "*TC: Small: Exploiting Software Elasticity for Automatic Software Self-Healing*", NSF Trustworthy Computing, CNS-09-14845, **\$450,000** (07/2009 - 06/2012)
  16. co-PI (with Steve Bellovin and Sal Stolfo), "*Pro-actively Removing the Botnet Threat*", Office of Naval Research (ONR), **\$294,625** (04/2009 - 09/2010)
  17. co-PI (with Simha Sethumadhavan and Sal Stolfo), "*SCOPS: Secure Cyber Operations and Parallelization Studies Cluster*", Air Force Office for Scientific Research (AFOSR), **\$650,000** (04/15/2009 - 04/14/2010)
  18. PI (co-PIs: Sal Stolfo), "*Program Whitelisting, Vulnerability Analytics and Risk Assessment*", Symantec (research gift), **\$65,000** (12/2008)
  19. co-PI (with Sal Stolfo), "*Automated Creation of Network and Content Traffic For the National Cyber Range*", DARPA/STO, **\$85,000** (01/01/2009 - 06/30/2011; part of a larger project)
  20. co-PI (with Steve Bellovin, Tal Malkin, and Sal Stolfo), "*Secure Encrypted Search*", IARPA, **\$648,787** (09/2008 - 02/2010)
  21. PI, "*Tracking Sensitive Information Flows in Modern Enterprises*", Intel (research gift), **\$64,000** (05/2008)
  22. PI, "*Privacy and Search: Having it Both Ways in Web Services*", Google (research gift), **\$50,000** (03/2008)
  23. PI (co-PI: Sal Stolfo), "*Continuation: Safe Browsing Through Web-based Application Communities*", Google (research gift), **\$50,000** (03/2008)
  24. co-PI (with Steve Bellovin, Vishal Misra, Henning Schulzrinne, Dan Rubenstein, Nick Maxemchuck), "*Zero Outage Dynamic Intrinsically Assurable Communities (ZODIAC)*", DARPA/STO, **\$835,357** (11/2007 - 05/2009; part of a larger project with Telcordia, Sparta, GMU, and the University of Pennsylvania)
  25. PI, "*Travel Supplement under the US/Japan Critical Infrastructure Protection Cooperation Program*", NSF CyberTrust, **\$38,640** (09/2007 - 08/2009)
  26. PI, "*PacketSpread: Practical Network Capabilities*", NSF CyberTrust, CNS-07-14277, **\$280,000** (09/2007 - 08/2010)
  27. PI, "*Integrated Enterprise Security Management*", NSF CyberTrust, CNS-07-14647,

- \$286,486** (08/2007 - 07/2009)
28. PI, "*Safe Browsing Through Web-based Application Communities*", NY State/Polytechnic CAT, **\$25,000** (06/2007 - 06/2009)
  29. PI, "*MURI: Foundational and Systems Support for Quantitative Trust Management*", Office of Naval Research (ONR), **\$750,000** (05/2007 - 04/2012; part of a larger project with the University of Pennsylvania and Georgia Institute of Technology)
  30. PI (co-PIs: Jason Nieh, Sal Stolfo), "*MURI: Autonomic Recovery of Enterprise-Wide Systems After Attack or Failure with Forward Correction*", Air Force Office of Scientific Research (AFOSR), **\$1,368,000** (05/2007 - 04/2012; part of a larger project with GMU and Penn State University)
  31. co-PI (with Sal Stolfo), "*Human Behavior, Insider Threat, and Awareness*", DHS/I3P, **\$616,442** (04/2007 - 03/2009)
  32. PI (co-PI: Sal Stolfo), "*Safe Browsing Through Web-based Application Communities*", Google (research gift), **\$50,000** (01/2007)
  33. PI (co-PI: Sal Stolfo), "*Supplement to Behavior-based Access Control and Communication in MANETs grant*", DARPA/IPTO and NRO, **\$96,627** (09/2006 - 07/2007)
  34. PI, "*Secure Overlay Services*", NY State/Polytechnic CAT, **\$10,000** (09/2006 - 06/2007)
  35. PI (co-PIs: Gail Kaiser, Sal Stolfo), "*Enabling Collaborative Self-healing Software Systems*", NSF CyberTrust, CNS-06-27473, **\$800,000** (09/2006 - 08/2010)
  36. PI (co-PI: Sal Stolfo), "*Behavior-based Access Control and Communication in MANETs*", DARPA/IPTO, **\$100,000** (07/2006 - 06/2007)
  37. co-PI (with Steve Bellovin and Sal Stolfo), "*Large-Scale System Defense*", DTO, **\$535,555** (07/2006 - 12/2007)
  38. PI, "*Active Decoys for Spyware*", NY State/Polytechnic CAT, **\$25,000** (06/2006 - 12/2007)
  39. PI, "*Retrofitting A Flow-oriented Paradigm in Commodity Operating Systems for High-Performance Computing*", NSF CPA, CCF-05-41093, **\$378,091** (01/2006 - 12/2008)
  40. co-PI (with Jason Nieh, Gail Kaiser), "*Broadening Participation in Research*", NSF BPC, **\$133,565** (09/2005 - 08/2006)
  41. PI, "*Secure Overlay Services*", NY State/Polytechnic CAT, **\$12,500** (09/2005 - 06/2006)
  42. co-PI (with Dan Rubenstein, Vishal Misra), "*Secure Overlay Services*", Intel Corp. (research gift), **\$75,000** (08/2005)
  43. PI, "*Snakeyes*", New York State Center for Advanced Technology, **\$14,999** (07/2005 - 06/2006)
  44. PI, "*Self-protecting Software*", Columbia Science and Technology Ventures (research gift), **\$65,000** (06/2005 - 09/2005)
  45. co-PI (with Gail Kaiser), "*Trustworthy Computing Curriculum Development*", Microsoft Research (research gift), **\$50,000** (12/2004 - 12/2005)
  46. co-PI (with Jason Nieh, Gail Kaiser), "*Secure Remote Computing Services*", NSF ITR, CNS-04-26623, **\$1,200,000** (09/2004 - 08/2009)
  47. PI, "*Secure Overlay Services*", NY State/Polytechnic CAT, **\$12,500** (09/2004 - 06/2005)
  48. co-PI (with Dan Rubenstein, Vishal Misra), "*Secure Overlay Services*", Intel Corp. (research gift), **\$90,000** (06/2004)
  49. co-PI (with Dan Rubenstein, Vishal Misra), "*Secure Overlay Services*", Intel Corp. (research gift), **\$120,000** (08/2003)
  50. PI (co-PIs: Dan Rubenstein, Vishal Misra), "*Secure Overlay Services*", Cisco Corp. (research gift), **\$76,000** (07/2003)
  51. co-PI (with Sal Stolfo, Tal Malkin, Vishal Misra), "*Distributed Intrusion Detection Feasibility Study*", Department of Defense, **\$300,000** (03/2003 - 03/2004)

52. PI, "*STRONGMAN*", DARPA/ATO, **\$23,782** (09/2002 - 08/2003; part of a larger project with the University of Pennsylvania)
53. PI, "*POSSE*", DARPA/ATO, \$16,341 (09/2002 - 08/2003; part of a larger project with the University of Pennsylvania)
54. PI, "*GRIDLOCK*", NSF Trusted Computing, CCR-TC-02-08972, **\$207,000** (07/2002 - 06/2005; part of a larger project with the University of Pennsylvania and Yale University)
55. PI (co-PIs: Dan Rubenstein, Vishal Misra), "*Secure Overlay Services*", Cisco Corp. (research gift), **\$70,000** (07/2002)
56. PI (co-PIs: Dan Rubenstein, Vishal Misra), "*Secure Overlay Services*", DARPA/ATO, **\$695,000** (06/2002 - 05/2004)
57. PI, "*Code Security Analysis Kit (CoSAK)*", DARPA/ATO, **\$37,000** (07/2001 - 06/2003; part of a larger project with Drexel University)

- **Total:** \$34,240,062
- **Total as PI:** \$20,625,555

#### Select Invited Talks

- "*Collaborative, Adaptive Software Defense*", invited talk, ONR Workshop on Host Computer Security, Chicago, IL, October 2010.
- "*Using Decoys to Identify Malicious Insiders*", invited talk, Computer Science Department, National University of Singapore, Singapore, August 2010.
- "*Behavior-based Access Control in Wired and Wireless Networks*", invited talk, 5<sup>th</sup> Ph.D. School on Security in Wireless Networking (SWING), Bertinoro, Italy, June/July 2010.
- "*MANET Security: Background and Distributed Defense*", invited talk, 5<sup>th</sup> Ph.D. School on Security in Wireless Networking (SWING), Bertinoro, Italy, June/July 2010.
- "*Detecting Insider Attackers*", invited talk, 5<sup>th</sup> Ph.D. School on Security in Wireless Networking (SWING), Bertinoro, Italy, June/July 2010.
- "*Self-healing and Collaborative Software Defenses*", invited talk, 5<sup>th</sup> Ph.D. School on Security in Wireless Networking (SWING), Bertinoro, Italy, June/July 2010.
- "*Voice over IP: Risks, Threats, and Vulnerabilities*", invited talk, 5<sup>th</sup> Ph.D. School on Security in Wireless Networking (SWING), Bertinoro, Italy, June/July 2010.
- "*Determining Device Trustworthiness in Heterogeneous Environments*", invited talk, Intel Workshop on Trust Evidence and End-to-end Trust in Heterogeneous Environments, Santa Clara, CA, May 2010.
- "*Moving Code: Instruction Set Randomization*", invited talk, IC Technical Exchange on Moving Target, Washington, DC, April 2010.
- "*Voice over IP: Risks, Threats and Vulnerabilities*", invited talk, AT&T Labs Research, Florham Park, NJ, April 2010.
- "*Voice over IP: Risks, Threats and Vulnerabilities*", keynote talk, 5<sup>th</sup> International Conference on Information Systems Security (ICISS), Kolkata, India, December 2009.
- "*Voice over IP: Risks, Threats and Vulnerabilities*", Cyber Infrastructure Protection (CIP) Conference, New York, June 2009.
- "*Voice over IP: Risks, Threats and Vulnerabilities*", keynote talk, Applied Cryptography and Network Security (ACNS) Conference, Paris, France, June 2009.
- "*Automatic Software Self-Healing: Present and Future*", keynote talk, European Workshop on Systems Security (EuroSec), Nuremberg, Germany, March 2009.
- "*VAMPIRE Project Overview*", Symantec Research Labs, Culver City, CA, March 2009.

- *"Survey of IMS/VoIP Security Work"*, Agence Nationale de Reserche (ANR), Paris, France, February 2009.
- *"Simulating a Global Passive Adversary for Attacking Tor-like Anonymity Systems"*, National Institute for Advanced Industrial Science and Technology (AIST), Japan, November 2008.
- *"Denial of Service Attacks and Resilient Overlay Networks"*, ENISA-FORTH Summer School on Network & Information Security, Heraklion, Greece, September 2008.
- *"von Neumann and the Current Computer Security Landscape"*, Onassis Foundation Lectures in Science, Heraklion, Greece, July 2008.
- *"Simulating a Global Passive Adversary for Attacking Tor-like Anonymity Systems"*, Institute of Computer Science/FORTH, Heraklion, Greece, July 2008.
- *"Race to the bottom: Malicious Hardware"*, 1<sup>st</sup> FORWARD Invitational Workshop for Identifying Emerging Threats in Information and Communication Technology Infrastructures, Goteborg, Sweden, April 2008.

### Publications

(Student co-authors are underlined.)

#### Patents

1. *"Microbilling using a trust management system"*  
Matthew A. Blaze, John Ioannidis, and Angelos D. Keromytis. U.S. Patent Number 7,996,325. Issued on August 9<sup>th</sup> 2011.
2. *"Methods, systems and media for software self-healing"*  
Michael E. Locasto, Angelos D. Keromytis, Salvatore J. Stolfo, Angelos Stavrou, Gabriela Cretu, Stylios Sidiroglou, Jason Nieh, and Oren Laadan. U.S. Patent Number 7,962,798. Issued on June 14<sup>th</sup>, 2011.
3. *"Systems and methods for detecting and inhibiting attacks using honeypots"*  
Stylios Sidiroglou, Angelos D. Keromytis, and Kostas G. Anagnostakis. U.S. Patent Number 7,904,959. Issued on March 8<sup>th</sup>, 2011.
4. *"Systems and methods for correlating and distributing intrusion alert information among collaborating computer systems"*  
Salvatore J. Stolfo, Angelos D. Keromytis, Vishal Misra, Michael Locasto, and Janak Parekh. U.S. Patent Number 7,784,097. Issued on August 24<sup>th</sup>, 2010.
5. *"Systems and methods for correlating and distributing intrusion alert information among collaborating computer systems"*  
Salvatore J. Stolfo, Tal Malkin, Angelos D. Keromytis, Vishal Misra, Michael Locasto, and Janak Parekh. U.S. Patent Number 7,779,463. Issued on August 17<sup>th</sup>, 2010.
6. *"Systems and methods for computing data transmission characteristics of a network path based on single-ended measurements"*  
Angelos D. Keromytis, Sambuddho Chakravarty, and Angelos Stavrou. U.S. Patent Number 7,660,261. Issued on February 9<sup>th</sup>, 2010.
7. *"Microbilling using a trust management system"*  
Matthew A. Blaze, John Ioannidis, and Angelos D. Keromytis. U.S. Patent Number 7,650,313. Issued on January 19<sup>th</sup> 2010.
8. *"Methods and systems for repairing applications"*  
Angelos D. Keromytis, Michael E. Locasto, and Stylios Sidiroglou. U.S. Patent Number 7,490,268. Issued on February 10<sup>th</sup> 2009.
9. *"System and method for microbilling using a trust management system"*

Matthew A. Blaze, John Ioannidis, and Angelos D. Keromytis. U.S. Patent Number 6,789,068. Issued on September 7<sup>th</sup> 2004.

10. "Secure and reliable bootstrap architecture"  
William A. Arbaugh, David J. Farber, Angelos D. Keromytis, and Jonathan M. Smith. U.S. Patent Number 6,185,678. Issued on February 6<sup>th</sup> 2001.

### Journal Publications

1. "A Comprehensive Survey of Voice over IP Security Research"  
Angelos D. Keromytis. To appear in the *IEEE Communications Surveys and Tutorials*.
2. "A System for Generating and Injecting Indistinguishable Network Decoys"  
Brian M. Bowen, Vasileios P. Kemerlis, Pratap Prabhu, Angelos D. Keromytis, and Salvatore J. Stolfo. To appear in the *Journal of Computer Security (JCS)*.
3. "The Efficient Dual Receiver Cryptosystem and Its Applications"  
Ted Diamant, Homin K. Lee, Angelos D. Keromytis, and Moti Yung. In *International Journal of Network Security (IJNS)*, vol 13, no. 3, pp. 135 - 151, November 2011.
4. "On the Infeasibility of Modeling Polymorphic Shellcode: Re-thinking the Role of Learning in Intrusion Detection Systems"  
Yingbo Song, Michael E. Locasto, Angelos Stavrou, Angelos D. Keromytis, and Salvatore J. Stolfo. In *Machine Learning Journal (MLJ)*, vol. 81, no. 2, pp. 179 - 205, November 2010.
5. "On The General Applicability of Instruction-Set Randomization"  
Stephen W. Boyd, Gaurav S. Kc, Michael E. Locasto, Angelos D. Keromytis, and Vassilis Prevelakis. In *IEEE Transactions on Dependable and Secure Computing (TDSC)*, vol. 7, no. 3, pp. 255 - 270, July - September 2010.
6. "Shadow Honey pots"  
Michalis Polychronakis, Periklis Akritidis, Stelios Sidiroglou, Kostas G. Anagnostakis, Angelos D. Keromytis, and Evangelos Markatos. In *International Journal of Computer and Network Security (IJCNS)*, vol. 2, no. 9, pp. 1 - 15, September 2010.
7. "Ethics in Security Vulnerability Research"  
Andrea M. Matwyshyn, Ang Cui, Salvatore J. Stolfo, and Angelos D. Keromytis. In *IEEE Security & Privacy Magazine*, vol. 8, no. 2, pp. 67 - 72, March/April 2010.
8. "Voice over IP Security: Research and Practice"  
Angelos D. Keromytis. In *IEEE Security & Privacy Magazine*, vol. 8, no. 2, pp. 76 - 78, March/April 2010.
9. "A Market-based Bandwidth Charging Framework"  
David Michael Turner, Vassilis Prevelakis, and Angelos D. Keromytis. In *ACM Transactions on Internet Technology (ToIT)*, vol. 10, no. 1, pp. 1 - 30, February 2010.
10. "A Look at VoIP Vulnerabilities"  
Angelos D. Keromytis. In *USENIX ;login: Magazine*, vol. 35, no. 1, pp. 41 - 50, February 2010.
11. "Designing Host and Network Sensors to Mitigate the Insider Threat"  
Brian M. Bowen, Malek Ben Salem, Shlomo Hershkop, Angelos D. Keromytis, and Salvatore J. Stolfo. In *IEEE Security & Privacy Magazine*, vol. 7, no. 6, pp. 22 - 29, November/December 2009.
12. "Elastic Block Ciphers: Method, Security and Instantiations"  
Debra L. Cook, Moti Yung, and Angelos D. Keromytis. In *Springer International Journal of Information Security (LIIS)*, vol 8, no. 3, pp. 211 - 231, June 2009.
13. "On the Deployment of Dynamic Taint Analysis for Application Communities"  
Hyung Chan Kim and Angelos D. Keromytis. In *IEICE Transactions*, vol. E92-D, no. 3, pp.



- 548 - 551, March 2009.
14. "Dynamic Trust Management"  
Matt Blaze, Sampath Kannan, Insup Lee, Oleg Sokolsky, Jonathan M. Smith, Angelos D. Keromytis, and Wenke Lee. In *IEEE Computer Magazine*, vol. 42, no. 2, pp. 44 - 52, February 2009.
  15. "Randomized Instruction Sets and Runtime Environments: Past Research and Future Directions"  
Angelos D. Keromytis. In *IEEE Security & Privacy Magazine*, vol. 7, no. 1, pp. 18 - 25, January/February 2009.
  16. "Anonymity in Wireless Broadcast Networks"  
Matt Blaze, John Ioannidis, Angelos D. Keromytis, Tal Malkin, and Avi Rubin. In *International Journal of Network Security (IJNS)*, vol. 8, no. 1, pp. 37 - 51, January 2009.
  17. "Decentralized Access Control in Networked File Systems"  
Stefan Miltchev, Jonathan M. Smith, Vassilis Prevelakis, Angelos D. Keromytis, and Sotiris Ioannidis. In *ACM Computing Surveys*, vol. 40, no. 3, pp. 10:1 - 10:30, August 2008.
  18. "Robust Reactions to Potential Day-Zero Worms through Cooperation and Validation"  
Kostas G. Anagnostakis, Michael Greenwald, Sotiris Ioannidis, and Angelos D. Keromytis. In *Springer International Journal of Information Security (IJIS), ISC 2006 Special Issue*, vol.6, no. 6, pp. 361 - 378, October 2007. (Extended version of the ISC 2006 paper.)
  19. "Requirements for Scalable Access Control and Security Management Architectures"  
Angelos D. Keromytis and Jonathan M. Smith. In *ACM Transactions on Internet Technology (ToIT)*, vol. 7, no. 2, pp. 1 - 22, May 2007.
  20. "Virtual Private Services: Coordinated Policy Enforcement for Distributed Applications"  
Sotiris Ioannidis, Steven M. Bellovin, John Ioannidis, Angelos D. Keromytis, Kostas G. Anagnostakis, and Jonathan M. Smith. In *International Journal of Network Security (IJNS)*, vol. 4, no. 1, pp. 69 - 80, January 2007.
  21. "Countering DDoS Attacks with Multi-path Overlay Networks"  
Angelos Stavrou and Angelos D. Keromytis. In *Information Assurance Technology Analysis Center (IATAC) Information Assurance Newsletter (IANewsletter)*, vol. 9, no. 3, pp. 26 - 30, Winter 2006. (Invited paper, based on the CCS 2005 paper.)
  22. "Conversion Functions for Symmetric Key Ciphers"  
Debra L. Cook and Angelos D. Keromytis. In *Journal of Information Assurance and Security (JIAS)*, vol. 1, no. 2, pp. 119 - 128, June 2006. (Extended version of the IAS 2005 paper.)
  23. "Execution Transactions for Defending Against Software Failures: Use and Evaluation"  
Stelios Sidiroglou and Angelos D. Keromytis. In *Springer International Journal of Information Security (IJIS)*, vol. 5, no. 2, pp. 77 - 91, April 2006. (Extended version of the ISC 2005 paper.)
  24. "Worm Propagation Strategies in an IPv6 Internet"  
Steven M. Bellovin, Bill Cheswick, and Angelos D. Keromytis. In *USENIX ;login*, vol. 31, no. 1, pp. 70 - 76, February 2006.
  25. "Cryptography As An Operating System Service: A Case Study"  
Angelos D. Keromytis, Theo de Raadt, Jason Wright, and Matthew Burnside. In *ACM Transactions on Computer Systems (ToCS)*, vol. 24, no. 1, pp. 1 - 38, February 2006. (Extended version of USENIX Technical 2003 paper.)
  26. "Countering Network Worms Through Automatic Patch Generation"  
Stelios Sidiroglou and Angelos D. Keromytis. In *IEEE Security & Privacy*, vol. 3, no. 6, pp. 41 - 49, November/December 2005.
  27. "WebSOS: An Overlay-based System For Protecting Web Servers From Denial of Service

*Attacks"*

- Angelos Stavrou, Debra L. Cook, William G. Morein, Angelos D. Keromytis, Vishal Misra, and Dan Rubenstein. In *Elsevier Journal of Computer Networks, special issue on Web and Network Security*, vol. 48, no. 5, pp. 781 - 807, August 2005. (Extended version of the CCS 2003 paper.)
28. "Hardware Support For Self-Healing Software Services"  
Stelios Sidiroglou, Michael E. Locasto, and Angelos D. Keromytis. In *ACM SIGARCH Computer Architecture News, Special Issue on Workshop on Architectural Support for Security and Anti-Virus (WASSA)*, vol. 33, no. 1, pp. 42 - 47, March 2005. Also appeared in the Proceedings of the *Workshop on Architectural Support for Security and Anti-Virus (WASSA)*, held in conjunction with the 11<sup>th</sup> International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-XI), pp. 37 - 43. October 2004, Boston, MA.
  29. "The Case For Crypto Protocol Awareness Inside The OS Kernel"  
Matthew Burnside and Angelos D. Keromytis. In *ACM SIGARCH Computer Architecture News, Special Issue on Workshop on Architectural Support for Security and Anti-Virus (WASSA)*, vol. 33, no. 1, pp. 58 - 64, March 2005. Also appeared in the Proceedings of the *Workshop on Architectural Support for Security and Anti-Virus (WASSA)*, held in conjunction with the 11<sup>th</sup> International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-XI), pp. 54 - 60. October 2004, Boston, MA.
  30. "Patch-on-Demand Saves Even More Time?"  
Angelos D. Keromytis. In *IEEE Computer*, vol. 37, no. 8, pp. 94 - 96, August 2004.
  31. "Just Fast Keying: Key Agreement In A Hostile Internet"  
William Aiello, Steven M. Bellovin, Matt Blaze, Ran Canetti, John Ioannidis, Angelos D. Keromytis, and Omer Reingold. In *ACM Transactions on Information and System Security (TISSEC)*, vol. 7, no. 2, pp. 1 - 32, May 2004. (Extended version of the CCS 2002 paper.)
  32. "SOS: An Architecture for Mitigating DDoS Attacks"  
Angelos D. Keromytis, Vishal Misra, and Dan Rubenstein. In *IEEE Journal on Selected Areas in Communications (JSAC), special issue on Recent Advances in Service Overlay Networks*, vol. 22, no. 1, pp. 176 - 188, January 2004. (Extended version of the SIGCOMM 2002 paper.)
  33. "A Secure PLAN"  
Michael Hicks, Angelos D. Keromytis, and Jonathan M. Smith. In *IEEE Transactions on Systems, Man, and Cybernetics (T-SMC) Part C: Applications and Reviews, Special issue on technologies promoting computational intelligence, openness and programmability in networks and Internet services: Part I*, vol. 33, no. 3, pp. 413 - 426, August 2003. (Extended version of the DANCE 2002 paper.)
  34. "Drop-in Security for Distributed and Portable Computing Elements"  
Vassilis Prevelakis and Angelos D. Keromytis. In *MCB Press Emerald Journal of Internet Research: Electronic Networking, Applications and Policy*, vol. 13, no. 2, pp. 107 - 115, 2003. (Extended version of the INC 2002 paper.)
  35. "Trust Management for IPsec"  
Matt Blaze, John Ioannidis, and Angelos D. Keromytis. In *ACM Transactions on Information and System Security (TISSEC)*, vol. 5, no. 2, pp. 1 - 24, May 2002. (Extended version of the NDSS 2001 paper.)
  36. "The Price of Safety in an Active Network"  
D. Scott Alexander, Paul B. Menage, Angelos D. Keromytis, William A. Arbaugh, Kostas G.

- Anagnostakis, and Jonathan M. Smith. In *Journal of Communications and Networks (JCN)*, special issue on programmable switches and routers, vol. 3, no. 1, pp. 4 - 18, March 2001. Older versions are available as *University of Pennsylvania Technical Report MS-CIS-99-04* and *University of Pennsylvania Technical Report MS-CIS-98-02*.
37. "Secure Quality of Service Handling (SQoSH)"  
D. Scott Alexander, William A. Arbaugh, Angelos D. Keromytis, Steve Muir, and Jonathan M. Smith. In *IEEE Communications Magazine*, vol. 38, no. 4, pp. 106 - 112, April 2000. An older version is available as *University of Pennsylvania Technical Report MS-CIS-99-05*.
  38. "Safety and Security of Programmable Network Infrastructures"  
D. Scott Alexander, William A. Arbaugh, Angelos D. Keromytis, and Jonathan M. Smith. In *IEEE Communications Magazine*, issue on Programmable Networks, vol. 36, no. 10, pp. 84 - 92, October 1998.
  39. "A Secure Active Network Environment Architecture"  
D. Scott Alexander, William A. Arbaugh, Angelos D. Keromytis, and Jonathan M. Smith. In *IEEE Network Magazine*, special issue on Active and Controllable Networks, vol. 12, no. 3, pp. 37 - 45, May/June 1998.
  40. "The SwitchWare Active Network Architecture"  
D. Scott Alexander, William A. Arbaugh, Michael Hicks, Pankaj Kakkar, Angelos D. Keromytis, Jonathan T. Moore, Carl A. Gunter, Scott M. Nettles, and Jonathan M. Smith. In *IEEE Network Magazine*, special issue on Active and Programmable Networks, vol. 12, no. 3, pp. 29 - 36, May/June 1998.

#### Peer-Reviewed Conference Proceedings

1. "A Multilayer Overlay Network Architecture for Enhancing IP Services Availability Against DoS"  
Dimitris Geneiatakis, Georgios Portokalidis, and Angelos D. Keromytis. To appear in the Proceedings of the 7<sup>th</sup> International Conference on Information Systems Security (ICISS). December 2011, Kolkata, India. (Acceptance rate: 22.8%)
2. "ROP Payload Detection Using Speculative Code Execution"  
Michalis Polychronakis and Angelos D. Keromytis. To appear in the Proceedings of the 6<sup>th</sup> International Conference on Malicious and Unwanted Software (MALWARE). October 2011, Fajardo, PR.
3. "Detecting Traffic Snooping in Tor Using Decoys"  
Sambuddho Chakravarty, Georgios Portokalidis, Michalis Polychronakis, and Angelos D. Keromytis. To appear in Proceedings of the 14<sup>th</sup> International Symposium on Recent Advances in Intrusion Detection (RAID). September 2011, Menlo Park, CA. (Acceptance rate: 23%)
4. "Measuring the Deployment Hiccups of DNSSEC"  
Vasilis Pappas and Angelos D. Keromytis. In Proceedings of the International Conference on Advances in Computing and Communications (ACC), Part III, pp. 44 - 54. July 2011, Kochi, India. (Acceptance rate: 39%)
5. "Misuse Detection in Consent-based Networks"  
Mansoor Alicherry and Angelos D. Keromytis. In Proceedings of the 9<sup>th</sup> International Conference on Applied Cryptography and Network Security (ACNS), pp. 38 - 56. June 2011, Malaga, Spain. (Acceptance rate: 18%)
6. "Retrofitting Security in COTS Software with Binary Rewriting"  
Padraig O'Sullivan, Kapil Anand, Aparna Kothan, Matthew Smithson, Rajeev Barua, and Angelos D. Keromytis. In Proceedings of the 26<sup>th</sup> IFIP International Information Security

- Conference (SEC)*, pp. 154 - 172. June 2011, Lucerne, Switzerland. (Acceptance rate: 24%)
7. *"Fast and Practical Instruction-Set Randomization for Commodity Systems"*  
Georgios Portokalidis and Angelos D. Keromytis. In Proceedings of the 26<sup>th</sup> Annual Computer Security Applications Conference (ACSAC), pp. 41 - 48. December 2010, Austin, TX. (Acceptance rate: 17%)
  8. *"An Adversarial Evaluation of Network Signaling and Control Mechanisms"*  
Kangkook Jee, Stelios Sidiroglou-Douskos, Angelos Stavrou, and Angelos D. Keromytis. In Proceedings of the 13<sup>th</sup> International Conference on Information Security and Cryptology (ICISC). December 2010, Seoul, Korea.
  9. *"Evaluation of a Spyware Detection System using Thin Client Computing"*  
Vasilis Pappas, Brian M. Bowen, and Angelos D. Keromytis. In Proceedings of the 13<sup>th</sup> International Conference on Information Security and Cryptology (ICISC), pp. 222 - 232. December 2010, Seoul, Korea.
  10. *"Crimeware Swindling without Virtual Machines"*  
Vasilis Pappas, Brian M. Bowen, and Angelos D. Keromytis. In Proceedings of the 13<sup>th</sup> Information Security Conference (ISC), pp. 196 - 202. October 2010, Boca Raton, FL. (Acceptance rate: 27.6%)
  11. *"iLeak: A Lightweight System for Detecting Inadvertent Information Leaks"*  
Vasileios P. Kemerlis, Vasilis Pappas, Georgios Portokalidis, and Angelos D. Keromytis. In Proceedings of the 6<sup>th</sup> European Conference on Computer Network Defense (EC2ND), pp. 21 - 28. October 2010, Berlin, Germany.
  12. *"Traffic Analysis Against Low-Latency Anonymity Networks Using Available Bandwidth Estimation"*  
Sambuddho Chakravarty, Angelos Stavrou, and Angelos D. Keromytis. In Proceedings of the 15<sup>th</sup> European Symposium on Research in Computer Security (ESORICS), pp. 249 - 267. September 2010, Athens, Greece. (Acceptance rate: 20%)
  13. *"BotSwindler: Tamper Resistant Injection of Believable Decoys in VM-Based Hosts for Crimeware Detection"*  
Brian M. Bowen, Pratap Prabhu, Vasileios P. Kemerlis, Stelios Sidiroglou, Angelos D. Keromytis, and Salvatore J. Stolfo. In Proceedings of the 13<sup>th</sup> International Symposium on Recent Advances in Intrusion Detection (RAID), pp. 118 - 137. September 2010, Ottawa, Canada. (Acceptance rate: 23.5%)
  14. *"An Analysis of Rogue AV Campaigns"*  
Marco Cova, Corrado Leita, Olivier Thonnard, Angelos D. Keromytis, and Marc Dacier. In Proceedings of the 13<sup>th</sup> International Symposium on Recent Advances in Intrusion Detection (RAID), pp. 442 - 463. September 2010, Ottawa, Canada. (Acceptance rate: 23.5%)
  15. *"DIPLOMA: Distributed Policy Enforcement Architecture for MANETs"*  
Mansoor Alicherry and Angelos D. Keromytis. In Proceedings of the 4<sup>th</sup> International Conference on Network and System Security (NSS), pp. 89 - 98. September 2010, Melbourne, Australia. (Acceptance rate: 26%)
  16. *"Automating the Injection of Believable Decoys to Detect Snooping" (Short Paper)*  
Brian M. Bowen, Vasileios Kemerlis, Pratap Prabhu, Angelos D. Keromytis, and Salvatore J. Stolfo. In Proceedings of the 3<sup>rd</sup> ACM Conference on Wireless Network Security (WiSec), pp. 81 - 86. March 2010, Hoboken, NJ. (Acceptance rate: 21%)
  17. *"BARTER: Behavior Profile Exchange for Behavior-Based Admission and Access Control in MANETs"*  
Vanessa Frias-Martinez, Salvatore J. Stolfo, and Angelos D. Keromytis. In Proceedings of the 5<sup>th</sup> International Conference on Information Systems Security (ICISS), pp. 193 - 207.

- December 2009, Kolkata, India. (Acceptance rate: 19.8%)
18. *"A Survey of Voice Over IP Security Research"*  
Angelos D. Keromytis. In Proceedings of the 5<sup>th</sup> International Conference on Information Systems Security (ICISS), pp. 1 - 17. December 2009, Kolkata, India. (Invited paper)
  19. *"A Network Access Control Mechanism Based on Behavior Profiles"*  
Vanessa Frias-Martinez, Joseph Sherrick, Salvatore J. Stolfo, and Angelos D. Keromytis. In Proceedings of the 25<sup>th</sup> Annual Computer Security Applications Conference (ACSAC), pp. 3 - 12. December 2009, Honolulu, HI. (Acceptance rate: 20%)
  20. *"Gone Rogue: An Analysis of Rogue Security Software Campaigns"*  
Marco Cova, Corrado Leita, Olivier Thonnard, Angelos D. Keromytis, and Marc Dacier. In Proceedings of the 5<sup>th</sup> European Conference on Computer Network Defense (EC2ND), pp. 1 - 3. November 2009, Milan, Italy. (Invited paper)
  21. *"Baiting Inside Attackers Using Decoy Documents"*  
Brian M. Bowen, Shlomo Hershkop, Angelos D. Keromytis, and Salvatore J. Stolfo. In Proceedings of the 5<sup>th</sup> International ICST Conference on Security and Privacy in Communication Networks (SecureComm), pp. 51 - 70. September 2009, Athens, Greece. (Acceptance rate: 25.3%)
  22. *"Deny-by-Default Distributed Security Policy Enforcement in Mobile Ad Hoc Networks (Short Paper)"*  
Mansoor Alicherry, Angelos D. Keromytis, and Angelos Stavrou. In Proceedings of the 5<sup>th</sup> International ICST Conference on Security and Privacy in Communication Networks (SecureComm), pp. 41 - 50. September 2009, Athens, Greece. (Acceptance rate: 34.7%)
  23. *"Adding Trust to P2P Distribution of Paid Content"*  
Alex Sherman, Angelos Stavrou, Jason Nieh, Angelos D. Keromytis, and Clifford Stein. In Proceedings of the 12<sup>th</sup> Information Security Conference (ISC), pp. 459 - 474. September 2009, Pisa, Italy. (Acceptance rate: 27.6%)
  24. *"A2M: Access-Assured Mobile Desktop Computing"*  
Angelos Stavrou, Ricardo A. Baratto, Angelos D. Keromytis, and Jason Nieh. In Proceedings of the 12<sup>th</sup> Information Security Conference (ISC), pp. 186 - 201. September 2009, Pisa, Italy. (Acceptance rate: 27.6%)
  25. *"F3ildCrypt: End-to-End Protection of Sensitive Information in Web Services"*  
Matthew Burnside and Angelos D. Keromytis. In Proceedings of the 12<sup>th</sup> Information Security Conference (ISC), pp. 491 - 506. September 2009, Pisa, Italy. (Acceptance rate: 27.6%)
  26. *"DoubleCheck: Multi-path Verification Against Man-in-the-Middle Attacks"*  
Mansoor Alicherry and Angelos D. Keromytis. In Proceedings of the IEEE Symposium on Computers and Communications (ISCC), pp. 557 - 563. July 2009, Sousse, Tunisia. (Acceptance rate: 36%)
  27. *"Voice over IP: Risks, Threats and Vulnerabilities"*  
Angelos D. Keromytis. In Proceedings (electronic) of the Cyber Infrastructure Protection (CIP) Conference. June 2009, New York, NY. (Invited paper)
  28. *"Capturing Information Flow with Concatenated Dynamic Taint Analysis"*  
Hyung Chan Kim, Angelos D. Keromytis, Michael Covington, and Ravi Sahita. In Proceedings of the 4<sup>th</sup> International Conference on Availability, Reliability and Security (ARES), pp. 355 - 362. March 2009, Fukuoka, Japan. (Acceptance rate: 25%)
  29. *"ASSURE: Automatic Software Self-healing Using REscue points"*  
Stelios Sidiroglou, Oren Laadan, Nico Viennot, Carlos-René Pérez, Angelos D. Keromytis, and Jason Nieh. In Proceedings of the 14<sup>th</sup> International Conference on Architectural Support

- for *Programming Languages and Operating Systems (ASPLOS)*, pp. 37 - 48. March 2009, Washington, DC. (Acceptance rate: 25.6%)
30. "Spectrogram: A Mixture-of-Markov-Chains Model for Anomaly Detection in Web Traffic" Yingbo Song, Angelos D. Keromytis, and Salvatore J. Stolfo. In Proceedings of the 16<sup>th</sup> *Internet Society (ISOC) Symposium on Network and Distributed Systems Security (SNDSS)*, pp. 121 - 135. February 2009, San Diego, CA. (Acceptance rate: 11.7%)
  31. "Constructing Variable-Length PRPs and SPRPs from Fixed-Length PRPs" Debra L. Cook, Moti Yung, and Angelos D. Keromytis. In Proceedings of the 4<sup>th</sup> *International Conference on Information Security and Cryptology (Inscrypt)*, pp. 157 - 180. December 2008, Beijing, China. (Acceptance rate: 17.5%)
  32. "Behavior-Profile Clustering for False Alert Reduction in Anomaly Detection Sensors" Vanessa Frias-Martinez, Salvatore J. Stolfo, and Angelos D. Keromytis. In Proceedings of the 24<sup>th</sup> *Annual Computer Security Applications Conference (ACSAC)*, pp. 367 - 376. December 2008, Anaheim, CA. (Acceptance rate: 24.2%)
  33. "Authentication on Untrusted Remote Hosts with Public-key Sudo" Matthew Burnside, Mack Lu, and Angelos D. Keromytis. In Proceedings of the 22<sup>nd</sup> *USENIX Large Installation Systems Administration (LISA) Conference*, pp. 103 - 107. November 2008, San Diego, CA.
  34. "Behavior-Based Network Access Control: A Proof-of-Concept" Vanessa Frias-Martinez, Salvatore J. Stolfo, and Angelos D. Keromytis. In Proceedings of the 11<sup>th</sup> *Information Security Conference (ISC)*, pp. 175 - 190. Taipei, Taiwan, September 2008. (Acceptance rate: 23.9%)
  35. "Path-based Access Control for Enterprise Networks" Matthew Burnside and Angelos D. Keromytis. In Proceedings of the 11<sup>th</sup> *Information Security Conference (ISC)*, pp. 191 - 203. Taipei, Taiwan, September 2008. (Acceptance rate: 23.9%)
  36. "Methods for Linear and Differential Cryptanalysis of Elastic Block Ciphers" Debra L. Cook, Moti Yung, and Angelos D. Keromytis. In Proceedings of the 13<sup>th</sup> *Australasian Conference on Information Security and Privacy (ACISP)*, pp. 187 - 202. July 2008, Wollongong, Australia. (Acceptance rate: 29.7%)
  37. "Pushback for Overlay Networks: Protecting against Malicious Insiders" Angelos Stavrou, Michael E. Locasto, and Angelos D. Keromytis. In Proceedings of the 6<sup>th</sup> *International Conference on Applied Cryptography and Network Security (ACNS)*, pp 39 - 54. June 2008, New York, NY. (Acceptance rate: 22.9%)
  38. "Casting out Demons: Sanitizing Training Data for Anomaly Sensors" Gabriela F. Cretu, Angelos Stavrou, Michael E. Locasto, Salvatore J. Stolfo, and Angelos D. Keromytis. In Proceedings of the *IEEE Symposium on Security & Privacy*, pp. 81 - 95. May 2008, Oakland, CA. (Acceptance rate: 11.2%)
  39. "Taming the Devil: Techniques for Evaluating Anonymized Network Data" Scott E. Coull, Charles V. Wright, Angelos D. Keromytis, Fabian Monrose, and Michael K. Reiter. In Proceedings of the 15<sup>th</sup> *Internet Society (ISOC) Symposium on Network and Distributed Systems Security (SNDSS)*, pp. 125 - 135. February 2008, San Diego, CA. (Acceptance rate: 17.8%)
  40. "SSARES: Secure Searchable Automated Remote Email Storage" Adam J. Aviv, Michael E. Locasto, Shaya Potter, and Angelos D. Keromytis. In Proceedings of the 23<sup>rd</sup> *Annual Computer Security Applications Conference (ACSAC)*, pp. 129 - 138. December 2007, Miami Beach, FL. (Acceptance rate: 22%)
  41. "On the Infeasibility of Modeling Polymorphic Shellcode"

- Yingbo Song, Michael E. Locasto, Angelos Stavrou, Angelos D. Keromytis, and Salvatore J. Stolfo. In Proceedings of the 13<sup>th</sup> ACM Conference on Computer and Communications Security (CCS), pp. 541 - 551. October/November 2007, Alexandria, VA. (Acceptance rate: 18.1%)
42. "Defending Against Next Generation Attacks Through Network/Endpoint Collaboration and Interaction"  
Spiros Antonatos, Michael E. Locasto, Stelios Sidiroglou, Angelos D. Keromytis, and Evangelos Markatos. In Proceedings of the 3<sup>rd</sup> European Conference on Computer Network Defense (EC2ND). October 2007, Heraclion, Greece. (Invited paper)
  43. "Elastic Block Ciphers in Practice: Constructions and Modes of Encryption"  
Debra L. Cook, Moti Yung, and Angelos D. Keromytis. In Proceedings of the 3<sup>rd</sup> European Conference on Computer Network Defense (EC2ND). October 2007, Heraclion, Greece.
  44. "The Security of Elastic Block Ciphers Against Key-Recovery Attacks"  
Debra L. Cook, Moti Yung, and Angelos D. Keromytis. In Proceedings of the 10<sup>th</sup> Information Security Conference (ISC), pp. 89 - 103. Valparaiso, Chile, October 2007. (Acceptance rate: 25%)
  45. "Characterizing Self-healing Software Systems"  
Angelos D. Keromytis. In Proceedings of the 4<sup>th</sup> International Conference on Mathematical Methods, Models and Architectures for Computer Networks Security (MMM-ACNS), pp. 22 - 33. September 2007, St. Petersburg, Russia. (Invited paper)
  46. "A Study of Malcode-Bearing Documents"  
Wei-Jen Li, Salvatore J. Stolfo, Angelos Stavrou, Elli Androulaki, and Angelos D. Keromytis. In Proceedings of the 4<sup>th</sup> GI International Conference on Detection of Intrusions & Malware, and Vulnerability Assessment (DIMVA), pp. 231 - 250. July 2007, Lucerne, Switzerland. (Acceptance rate: 21%)
  47. "From STEM to SEAD: Speculative Execution for Automated Defense"  
Michael E. Locasto, Angelos Stavrou, Gabriela F. Cretu, and Angelos D. Keromytis. In Proceedings of the USENIX Annual Technical Conference, pp. 219 - 232. June 2007, Santa Clara, CA. (Acceptance rate: 18.75%)
  48. "Using Rescue Points to Navigate Software Recovery (Short Paper)"  
Stelios Sidiroglou, Oren Laadan, Angelos D. Keromytis, and Jason Nieh. In Proceedings of the IEEE Symposium on Security & Privacy, pp. 273 - 278. May 2007, Oakland, CA. (Acceptance rate: 8.3%)
  49. "Mediated Overlay Services (MOSES): Network Security as a Composable Service"  
Stelios Sidiroglou, Angelos Stavrou, and Angelos D. Keromytis. In Proceedings of the IEEE Sarnoff Symposium. May 2007, Princeton, NJ. (Invited paper)
  50. "Elastic Block Ciphers: The Basic Design"  
Debra L. Cook, Moti Yung, and Angelos D. Keromytis. In Proceedings of the 2<sup>nd</sup> ACM Symposium on InformAtion, Computer and Communications Security (ASIACCS), pp. 350 - 355. March 2007, Singapore.
  51. "Robust Reactions to Potential Day-Zero Worms through Cooperation and Validation"  
Kostas G. Anagnostakis, Michael B. Greenwald, Sotiris Ioannidis, and Angelos D. Keromytis. In Proceedings of the 9<sup>th</sup> Information Security Conference (ISC), pp. 427 - 442. August/September 2006, Samos, Greece. (Acceptance rate: 20.2%)
  52. "Low Latency Anonymity with Mix Rings"  
Matthew Burnside and Angelos D. Keromytis. In Proceedings of the 9<sup>th</sup> Information Security Conference (ISC), pp. 32 - 45. August/September 2006, Samos, Greece. (Acceptance rate: 20.2%)

53. *"W3Bcrypt: Encryption as a Stylesheet"*  
Angelos Stavrou, Michael E. Locasto, and Angelos D. Keromytis. In Proceedings of the 4<sup>th</sup> International Conference on Applied Cryptography and Network Security (ACNS), pp. 349 - 364. June 2006, Singapore.
54. *"Software Self-Healing Using Collaborative Application Communities"*  
Michael E. Locasto, Stelios Sidiroglou, and Angelos D. Keromytis. In Proceedings of the 13<sup>th</sup> Internet Society (ISOC) Symposium on Network and Distributed Systems Security (SNDSS), pp. 95 - 106. February 2006, San Diego, CA. (Acceptance rate: 13.6%)
55. *"Remotely Keyed Cryptographics: Secure Remote Display Access Using (Mostly) Untrusted Hardware"*  
Debra L. Cook, Ricardo A. Baratto, and Angelos D. Keromytis. In Proceedings of the 7<sup>th</sup> International Conference on Information and Communications Security (ICICS), pp. 363 - 375. December 2005, Beijing, China. (Acceptance rate: 17.4%)
56. *"e-NeXSh: Achieving an Effectively Non-Executable Stack and Heap via System-Call Policing"*  
Gaurav S. Kc and Angelos D. Keromytis. In Proceedings of the 21<sup>st</sup> Annual Computer Security Applications Conference (ACSAC), pp. 259 - 273. December 2005, Tucson, AZ. (Acceptance rate: 19.6%)
57. *"Action Amplification: A New Approach To Scalable Administration"*  
Kostas G. Anagnostakis and Angelos D. Keromytis. In Proceedings of the 13<sup>th</sup> IEEE International Conference on Networks (ICON), vol. 2, pp. 862 - 867. November 2005, Kuala Lumpur, Malaysia.
58. *"A Repeater Encryption Unit for IPv4 and IPv6"*  
Norimitsu Nagashima and Angelos D. Keromytis. In Proceedings of the 13<sup>th</sup> IEEE International Conference on Networks (ICON), vol. 1, pp. 335 - 340. November 2005, Kuala Lumpur, Malaysia.
59. *"Countering DoS Attacks With Stateless Multipath Overlays"*  
Angelos Stavrou and Angelos D. Keromytis. In Proceedings of the 12<sup>th</sup> ACM Conference on Computer and Communications Security (CCS), pp. 249 - 259. November 2005, Alexandria, VA. (Acceptance rate: 15.2%)
60. *"A Dynamic Mechanism for Recovering from Buffer Overflow Attacks"*  
Stelios Sidiroglou, Giannis Giovanidis, and Angelos D. Keromytis. In Proceedings of the 8<sup>th</sup> Information Security Conference (ISC), pp. 1 - 15. September 2005, Singapore. (Acceptance rate: 14%)
61. *"gore: Routing-Assisted Defense Against DDoS Attacks"*  
Stephen T. Chou, Angelos Stavrou, John Ioannidis, and Angelos D. Keromytis. In Proceedings of the 8<sup>th</sup> Information Security Conference (ISC), pp. 179 - 193. September 2005, Singapore. (Acceptance rate: 14%)
62. *"FLIPS: Hybrid Adaptive Intrusion Prevention"*  
Michael E. Locasto, Ke Wang, Angelos D. Keromytis, and Salvatore J. Stolfo. In Proceedings of the 8<sup>th</sup> International Symposium on Recent Advances in Intrusion Detection (RAID), pp. 82 - 101. September 2005, Seattle, WA. (Acceptance rate: 20.4%)
63. *"Detecting Targeted Attacks Using Shadow Honey Pots"*  
Kostas G. Anagnostakis, Stelios Sidiroglou, Periklis Akritidis, Konstantinos Xinidis, Evangelos Markatos, and Angelos D. Keromytis. In Proceedings of the 14<sup>th</sup> USENIX Security Symposium, pp. 129 - 144. August 2005, Baltimore, MD. (Acceptance rate: 14%)
64. *"The Bandwidth Exchange Architecture"*  
David Michael Turner, Vassilis Prevelakis, and Angelos D. Keromytis. In Proceedings of the



- 10<sup>th</sup> IEEE Symposium on Computers and Communications (ISCC), pp. 939 - 944. June 2005, Cartagena, Spain.
65. "An Email Worm Vaccine Architecture"  
Stelios Sidiroglou, John Ioannidis, Angelos D. Keromytis, and Salvatore J. Stolfo. In Proceedings of the 1<sup>st</sup> Information Security Practice and Experience Conference (ISPEC), pp. 97 - 108. April 2005, Singapore.
  66. "Building a Reactive Immune System for Software Services"  
Stelios Sidiroglou, Michael E. Locasto, Stephen W. Boyd, and Angelos D. Keromytis. In Proceedings of the USENIX Annual Technical Conference, pp. 149 - 161. April 2005, Anaheim, CA. (Acceptance rate: 20.3%)
  67. "Conversion and Proxy Functions for Symmetric Key Ciphers"  
Debra L. Cook and Angelos D. Keromytis. In Proceedings of the IEEE International Conference on Information Technology: Coding and Computing (ITCC), Information and Security (IAS) Track, pp. 662 - 667. April 2005, Las Vegas, NV.
  68. "The Effect of DNS Delays on Worm Propagation in an IPv6 Internet"  
Abhinav Kamra, Hanhua Feng, Vishal Misra, and Angelos D. Keromytis. In Proceedings of IEEE INFOCOM, vol. 4, pp. 2405 - 2414. March 2005, Miami, FL. (Acceptance rate: 17%)
  69. "MOVE: An End-to-End Solution To Network Denial of Service"  
Angelos Stavrou, Angelos D. Keromytis, Jason Nieh, Vishal Misra, and Dan Rubenstein. In Proceedings of the 12<sup>th</sup> Internet Society (ISOC) Symposium on Network and Distributed Systems Security (SNDSS), pp. 81 - 96. February 2005, San Diego, CA. (Acceptance rate: 12.9%)
  70. "CryptoGraphics: Secret Key Cryptography Using Graphics Cards"  
Debra L. Cook, John Ioannidis, Angelos D. Keromytis, and Jake Luck. In Proceedings of the RSA Conference, Cryptographer's Track (CT-RSA), pp. 334 - 350. February 2005, San Francisco, CA.
  71. "The Dual Receiver Cryptogram and Its Applications"  
Ted Diament, Homin K. Lee, Angelos D. Keromytis, and Moti Yung. In Proceedings of the 11<sup>th</sup> ACM Conference on Computer and Communications Security (CCS), pp. 330 - 343. October 2004, Washington, DC. (Acceptance rate: 13.9%)
  72. "Hydan: Hiding Information in Program Binaries"  
Rakan El-Khalil and Angelos D. Keromytis. In Proceedings of the 6<sup>th</sup> International Conference on Information and Communications Security (ICICS), pp. 187 - 199. October 2004, Malaga, Spain. (Acceptance rate: 16.9%)
  73. "Recursive Sandboxes: Extending Systrace To Empower Applications"  
Aleksey Kurchuk and Angelos D. Keromytis. In Proceedings of the 19<sup>th</sup> IFIP International Information Security Conference (SEC), pp. 473 - 487. August 2004, Toulouse, France. (Acceptance rate: 22%)
  74. "SQLrand: Preventing SQL Injection Attacks"  
Stephen W. Boyd and Angelos D. Keromytis. In Proceedings of the 2<sup>nd</sup> International Conference on Applied Cryptography and Network Security (ACNS), pp. 292 - 302. June 2004, Yellow Mountain, China. (Acceptance rate: 12.1%)
  75. "CamouflageFS: Increasing the Effective Key Length in Cryptographic Filesystems on the Cheap"  
Michael E. Locasto and Angelos D. Keromytis. In Proceedings of the 2<sup>nd</sup> International Conference on Applied Cryptography and Network Security (ACNS), pp. 1 - 15. June 2004, Yellow Mountain, China. (Acceptance rate: 12.1%)
  76. "A Pay-per-Use DoS Protection Mechanism For The Web"

- Angelos Stavrou, John Ioannidis, Angelos D. Keromytis, Vishal Misra, and Dan Rubenstein. In Proceedings of the 2<sup>nd</sup> *International Conference on Applied Cryptography and Network Security (ACNS)*, pp. 120 - 134. June 2004, Yellow Mountain, China. (Acceptance rate: 12.1%)
77. *"Dealing with System Monocultures"*  
Angelos D. Keromytis and Vassilis Prevelakis. In Proceedings (electronic) of the *NATO Information Systems Technology (IST) Panel Symposium on Adaptive Defense in Unclassified Networks*. April 2004, Toulouse, France.
  78. *"Managing Access Control in Large Scale Heterogeneous Networks"*  
Angelos D. Keromytis, Kostas G. Anagnostakis, Sotiris Ioannidis, Michael Greenwald, and Jonathan M. Smith. In Proceedings (electronic) of the *NATO NC3A Symposium on Interoperable Networks for Secure Communications (INSC)*. November 2003, The Hague, Netherlands.
  79. *"Countering Code-Injection Attacks With Instruction-Set Randomization"*  
Gaurav S. Kc, Angelos D. Keromytis, and Vassilis Prevelakis. In Proceedings of the 10<sup>th</sup> *ACM International Conference on Computer and Communications Security (CCS)*, pp. 272 - 280. October 2003, Washington, DC. (Acceptance rate: 13.8%)
  80. *"Using Graphic Turing Tests to Counter Automated DDoS Attacks Against Web Servers"*  
William G. Morein, Angelos Stavrou, Debra L. Cook, Angelos D. Keromytis, Vishal Misra, and Dan Rubenstein. In Proceedings of the 10<sup>th</sup> *ACM International Conference on Computer and Communications Security (CCS)*, pp. 8 - 19. October 2003, Washington, DC. (Acceptance rate: 13.8%)
  81. *"EasyVPN: IPsec Remote Access Made Easy"*  
Mark C. Benvenuto and Angelos D. Keromytis. In Proceedings of the 17<sup>th</sup> *USENIX Large Installation Systems Administration (LISA) Conference*, pp. 87 - 93. October 2003, San Diego, CA. (Acceptance rate: 25%)
  82. *"A Cooperative Immunization System for an Untrusting Internet"*  
Kostas G. Anagnostakis, Michael B. Greenwald, Sotiris Ioannidis, Angelos D. Keromytis, and Dekai Li. In Proceedings of the 11<sup>th</sup> *IEEE International Conference on Networks (ICON)*, pp. 403 - 408. September/October 2003, Sydney, Australia.
  83. *"Accelerating Application-Level Security Protocols"*  
Matthew Burnside and Angelos D. Keromytis. In Proceedings of the 11<sup>th</sup> *IEEE International Conference on Networks (ICON)*, pp. 313 - 318. September/October 2003, Sydney, Australia.
  84. *"WebSOS: Protecting Web Servers From DDoS Attacks"*  
Debra L. Cook, William G. Morein, Angelos D. Keromytis, Vishal Misra, and Dan Rubenstein. In Proceedings of the 11<sup>th</sup> *IEEE International Conference on Networks (ICON)*, pp. 455 - 460. September/October 2003, Sydney, Australia.
  85. *"TAPI: Transactions for Accessing Public Infrastructure"*  
Matt Blaze, John Ioannidis, Sotiris Ioannidis, Angelos D. Keromytis, Pekka Nikander, and Vassilis Prevelakis. In Proceedings of the 8<sup>th</sup> *IFIP Personal Wireless Communications (PWC) Conference*, pp. 90 - 100. September 2003, Venice, Italy.
  86. *"Tagging Data In The Network Stack: mbuf\_tags"*  
Angelos D. Keromytis. In Proceedings of the *USENIX BSD Conference (BSDCon)*, pp. 125 - 131. September 2003, San Mateo, CA.
  87. *"The Design of the OpenBSD Cryptographic Framework"*  
Angelos D. Keromytis, Jason L. Wright, and Theo de Raadt. In Proceedings of the *USENIX Annual Technical Conference*, pp. 181 - 196. June 2003, San Antonio, TX. (Acceptance rate: 23%)

88. *"Secure and Flexible Global File Sharing"*  
Stefan Miltchev, Vassilis Prevelakis, Sotiris Ioannidis, John Ioannidis, Angelos D. Keromytis, and Jonathan M. Smith. In Proceedings of the *USENIX Annual Technical Conference, Freenix Track*, pp. 165 - 178. June 2003, San Antonio, TX.
89. *"Experience with the KeyNote Trust Management System: Applications and Future Directions"*  
Matt Blaze, John Ioannidis, and Angelos D. Keromytis. In Proceedings of the *1<sup>st</sup> International Conference on Trust Management*, pp. 284 - 300. May 2003, Heraclion, Greece.
90. *"The STRONGMAN Architecture"*  
Angelos D. Keromytis, Sotiris Ioannidis, Michael B. Greenwald, and Jonathan M. Smith. In Proceedings of the *3<sup>rd</sup> DARPA Information Survivability Conference and Exposition (DISCEX III)*, volume 1, pp. 178 - 188. April 2003, Washington, DC.
91. *"Efficient, DoS-Resistant, Secure Key Exchange for Internet Protocols"*  
William Aiello, Steven M. Bellovin, Matt Blaze, Ran Canetti, John Ioannidis, Angelos D. Keromytis, and Omer Reingold. In Proceedings of the *9<sup>th</sup> ACM International Conference on Computer and Communications Security (CCS)*, pp. 48 - 58. November 2002, Washington, DC. (Acceptance rate: 17.6%)
92. *"Secure Overlay Services"*  
Angelos D. Keromytis, Vishal Misra, and Dan Rubenstein. In Proceedings of the *ACM SIGCOMM Conference*, pp. 61 - 72. August 2002, Pittsburgh, PA. Also available through the *ACM Computer Communications Review (SIGCOMM Proceedings)*, vol. 32, no. 4, October 2002. (Acceptance rate: 8.3%)
93. *"Using Overlays to Improve Network Security"*  
Angelos D. Keromytis, Vishal Misra, and Dan Rubenstein. In Proceedings of the *ITCom Conference*, special track on *Scalability and Traffic Control in IP Networks*, pp. 245 - 254. July/August 2002, Boston, MA. (Invited paper)
94. *"Designing an Embedded Firewall/VPN Gateway"*  
Vassilis Prevelakis and Angelos D. Keromytis. In Proceedings of the *International Network Conference (INC)*, pp. 313 - 322. July 2002, Plymouth, England. (Best Paper Award)
95. *"A Study of the Relative Costs of Network Security Protocols"*  
Stefan Miltchev, Sotiris Ioannidis, and Angelos D. Keromytis. In Proceedings of the *USENIX Annual Technical Conference, Freenix Track*, pp. 41 - 48. June 2002, Monterey, CA.
96. *"A Secure Plan (Extended Version)"*  
Michael W. Hicks, Angelos D. Keromytis, and Jonathan M. Smith. In Proceedings of the *DARPA Active Networks Conference and Exposition (DANCE)*, pp. 224 - 237. May 2002, San Francisco, CA. (Extended version of the paper *IWAN 1999 paper*.)
97. *"Fileteller: Paying and Getting Paid for File Storage"*  
John Ioannidis, Sotiris Ioannidis, Angelos D. Keromytis, and Vassilis Prevelakis. In Proceedings of the *6<sup>th</sup> Financial Cryptography (FC) Conference*, pp. 282 - 299. March 2002, Bermuda. (Acceptance rate: 25.6%)
98. *"Offline Micropayments without Trusted Hardware"*  
Matt Blaze, John Ioannidis, and Angelos D. Keromytis. In Proceedings of the *5<sup>th</sup> Financial Cryptography (FC) Conference*, pp. 21 - 40. February 2001, Cayman Islands.
99. *"Trust Management for IPsec"*  
Matt Blaze, John Ioannidis, and Angelos D. Keromytis. In Proceedings of the *8<sup>th</sup> Internet Society (ISOC) Symposium on Network and Distributed Systems Security (SNDSS)*, pp. 139 - 151. February 2001, San Diego, CA. (Acceptance rate: 24%)

100. *"Implementing a Distributed Firewall"*  
Sotiris Ioannidis, Angelos D. Keromytis, Steven M. Bellovin, and Jonathan M. Smith. In Proceedings of the 7<sup>th</sup> ACM International Conference on Computer and Communications Security (CCS), pp. 190 - 199. November 2000, Athens, Greece. (Acceptance rate: 21.4%)
101. *"Implementing Internet Key Exchange (IKE)"*  
Niklas Hallqvist and Angelos D. Keromytis. In Proceedings of the USENIX Annual Technical Conference, Freenix Track, pp. 201 - 214. June 2000, San Diego, CA.
102. *"Transparent Network Security Policy Enforcement"*  
Angelos D. Keromytis and Jason Wright. In Proceedings of the USENIX Annual Technical Conference, Freenix Track, pp. 215 - 226. June 2000, San Diego, CA.
103. *"Cryptography in OpenBSD: An Overview"*  
Theo de Raadt, Niklas Hallqvist, Artur Grabowski, Angelos D. Keromytis, and Niels Provos. In Proceedings of the USENIX Annual Technical Conference, Freenix Track, pp. 93 - 101. June 1999, Monterey, CA.
104. *"DHCP++: Applying an efficient implementation method for fail-stop cryptographic protocols"*  
William A. Arbaugh, Angelos D. Keromytis, and Jonathan M. Smith. In Proceedings of the IEEE Global Internet (GlobeCom), pp. 59 - 65. November 1998, Sydney, Australia.
105. *"Automated Recovery in a Secure Bootstrap Process"*  
William A. Arbaugh, Angelos D. Keromytis, David J. Farber, and Jonathan M. Smith. In Proceedings of the 5<sup>th</sup> Internet Society (ISOC) Symposium on Network and Distributed System Security (SNDSS), pp. 155 - 167. March 1998, San Diego, CA. An older version is available as University of Pennsylvania Technical Report MS-CIS-97-13.
106. *"Implementing IPsec"*  
Angelos D. Keromytis, John Ioannidis, and Jonathan M. Smith. In Proceedings of the IEEE Global Internet (GlobeCom), pp. 1948 - 1952. November 1997, Phoenix, AZ.

#### **Books/Book Chapters**

1. *"Voice over IP Security: A Comprehensive Survey of Vulnerabilities and Academic Research"*  
Angelos D. Keromytis. Springer Briefs, ISBN 978-1-4419-9865-1, April 2011.
2. *"Buffer Overflow Attacks"*  
Angelos D. Keromytis. In *Encyclopedia of Cryptography and Security, 2<sup>nd</sup> Edition*. Springer, 2011.
3. *"Network Bandwidth Denial of Service (DoS)"*  
Angelos D. Keromytis. In *Encyclopedia of Cryptography and Security, 2<sup>nd</sup> Edition*. Springer, 2011.
4. *"Monitoring Technologies for Mitigating Insider Threats"*  
Brian M. Bowen, Malek Ben Salem, Angelos D. Keromytis, and Salvatore J. Stolfo. In *Insider Threats in Cyber Security and Beyond*, Matt Bishop, Dieter Gollman, Jeffrey Hunker, and Christian Probst (editors), pp. 197 - 218. Springer, 2010.
5. *"Voice over IP: Risks, Threats, and Vulnerabilities"*  
Angelos D. Keromytis. In *Cyber Infrastructure Security*, Tarek Saadawi and Louis Jordan (editors). Strategic Study Institute (SSI), 2010.
6. *Proceedings of the 2008 New Security Paradigms Workshop (NSPW)*  
Angelos D. Keromytis, Anil Somayaji, and M. Hossain Heydari (editors).
7. *Proceedings of the 6<sup>th</sup> International Conference on Applied Cryptography and Network Security (ACNS)*

- Steven M. Bellovin, Rosario Gennaro, Angelos D. Keromytis, and Moti Yung (editors). Lecture Notes in Computer Science (LNCS). Springer, 2008.
8. *"Insider Attack and Cyber Security: Beyond the Hacker"*  
Salvatore J. Stolfo, Steven M. Bellovin, Angelos D. Keromytis, Sara Sinclair, and Sean W. Smith (editors). Advances in Information Security Series, ISBN 978-0387773216. Springer, 2008.
  9. *Proceedings of the 2007 New Security Paradigms Workshop (NSPW)*  
Kostantin Beznosov (Editor), Angelos D. Keromytis (editor), and M. Hossain Heydari (Editor).
  10. *"The Case for Self-Healing Software"*  
Angelos D. Keromytis. In *Aspects of Network and Information Security: Proceedings NATO Advanced Studies Institute (ASI) on Network Security and Intrusion Detection, held in Nork, Yerevan, Armenia, October 2006*, E. Haroutunian, E. Kranakis, and E. Shahbazian (editors). IOS Press, 2007. (By invitation, as part of the NATO ASI on Network Security, October 2005.)
  11. *"Designing Firewalls: A Survey"*  
Angelos D. Keromytis and Vassilis Prevelakis. In *Network Security: Current Status and Future Directions*, Christos Douligeris and Dimitrios N. Serpanos (editors), pp. 33 - 49. Wiley - IEEE Press, April 2007.
  12. *"Composite Hybrid Techniques for Defending against Targeted Attacks"*  
Stelios Sidiroglou and Angelos D. Keromytis. In *Malware Detection*, vol. 27 of Advances in Information Security Series, Mihai Christodorescu, Somesh Jha, Douglas Maughan, Dawn Song, and Cliff Wang (editors). Springer, October 2006. (By invitation, as part of the ARO/DHS 2005 Workshop on Malware Detection.)
  13. *"Trusted computing platforms and secure Operating Systems"*  
Angelos D. Keromytis. In *Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft*, Markus Jakobsson and Steven Myers (editors), pp. 387 - 405. Wiley, 2006.
  14. *"CryptoGraphics: Exploiting Graphics Cards for Security"*  
Debra Cook and Angelos D. Keromytis. Advances in Information Security Series, ISBN 0-387-29015-X. Springer, 2006.
  15. *Proceedings of the 3<sup>rd</sup> Workshop on Rapid Malcode (WORM)*  
Angelos D. Keromytis (editor). ACM Press, 2005.
  16. *Proceedings of the 3<sup>rd</sup> International Conference on Applied Cryptography and Network Security (ACNS)*  
John Ioannidis, Angelos D. Keromytis, and Moti Yung (editors). Lecture Notes in Computer Science (LNCS) 3531. Springer, 2005.
  17. *"Distributed Trust"*  
John Ioannidis and Angelos D. Keromytis. In *Practical Handbook of Internet Computing*, Munindar Singh (editor), pp. 47/1 - 47/16. CRC Press, 2004.
  18. *"Experiences Enhancing Open Source Security in the POSSE Project"*  
Jonathan M. Smith, Michael B. Greenwald, Sotiris Ioannidis, Angelos D. Keromytis, Ben Laurie, Douglas Maughan, Dale Rahn, and Jason L. Wright. In *Free/Open Source Software Development*, Stefan Koch (editor), pp. 242 - 257. Idea Group Publishing, 2004. Also re-published in *Global Information Technologies: Concepts, Methodologies, Tools, and Applications*, Felix B. Tan (editor), pp. 1587 - 1598. Idea Group Publishing, 2007.
  19. *"STRONGMAN: A Scalable Solution to Trust Management in Networks"*  
Angelos D. Keromytis. Ph.D. Thesis, University of Pennsylvania, November 2001.

20. *"The Role of Trust Management in Distributed Systems Security"*  
Matt Blaze, Joan Feigenbaum, John Ioannidis, and Angelos D. Keromytis. In *Secure Internet Programming: Issues in Distributed and Mobile Object Systems*, Jan Vitek and Christian Jensen (editors), pp. 185 - 210. Springer-Verlag Lecture Notes in Computer Science *State-of-the-Art* series, 1999.
21. *"Security in Active Networks"*  
D. Scott Alexander, William A. Arbaugh, Angelos D. Keromytis, and Jonathan M. Smith. In *Secure Internet Programming: Issues in Distributed and Mobile Object Systems*, Jan Vitek and Christian Jensen (editors), pp. 433 - 451. Springer-Verlag Lecture Notes in Computer Science *State-of-the-Art* series, 1999.

## Workshops

1. *"REASSURE: A Self-contained Mechanism for Healing Software Using Rescue Points"*  
Georgios Portokalidis and Angelos D. Keromytis. To appear in the Proceedings of the 6<sup>th</sup> *International Workshop on Security (IWSEC)*. November 2011, Tokyo, Japan.
2. *"Taint-Exchange: a Generic System for Cross-process and Cross-host Taint Tracking"*  
Angeliki Zavou, Georgios Portokalidis, and Angelos D. Keromytis. To appear in the Proceedings of the 6<sup>th</sup> *International Workshop on Security (IWSEC)*. November 2011, Tokyo, Japan.
3. *"The MINESTRONE Architecture: Combining Static and Dynamic Analysis Techniques for Software Security"*  
Angelos D. Keromytis, Salvatore J. Stolfo, Junfeng Yang, Angelos Stavrou, Anup Ghosh, Dawson Engler, Marc Dacier, Matthew Elder, and Darrell Kienzle. In Proceedings of the 1<sup>st</sup> *Workshop on Systems Security (SysSec)*. July 2011, Amsterdam, Netherlands.
4. *"The SPARCHS Project: Hardware Support for Software Security"*  
Simha Sethumadhavan, Salvatore J. Stolfo, David August, Angelos D. Keromytis, and Junfeng Yang. In Proceedings of the 1<sup>st</sup> *Workshop on Systems Security (SysSec)*. July 2011, Amsterdam, Netherlands.
5. *"Towards a Forensic Analysis for Multimedia Communication Services"*  
Dimitris Geneiatakis and Angelos D. Keromytis. In Proceedings of the 7<sup>th</sup> *International Symposium on Frontiers in Networking with Applications (FINA)*, pp. 424 - 429. March 2011, Biopolis, Singapore.
6. *"Security Research with Human Subjects: Informed Consent, Risk, and Benefits"*  
Maritza Johnson, Steven M. Bellovin, and Angelos D. Keromytis. In Proceedings of the 2<sup>nd</sup> *Workshop on Ethics in Computer Security Research (WECSR)*. March 2011, Saint Lucia.
7. *"Global ISR: Toward a Comprehensive Defense Against Unauthorized Code Execution"*  
Georgios Portokalidis and Angelos D. Keromytis. In Proceedings of the *ARO Workshop on Moving Target Defense*. October 2010, Fairfax, VA.
8. *"Securing MANET Multicast Using DIPLOMA"*  
Mansoor Alicherry and Angelos D. Keromytis. In Proceedings of the 5<sup>th</sup> *International Workshop on Security (IWSEC)*, pp. 232 - 250. November 2010, Kobe, Japan. (Acceptance rate: 29%)
9. *"Evaluating a Collaborative Defense Architecture for MANETs"*  
Mansoor Alicherry, Angelos Stavrou, and Angelos D. Keromytis. In Proceedings (electronic) of the *IEEE Workshop on Collaborative Security Technologies (CoSec)*, pp. 37 - 42. December 2009, Bangalore, India. (Acceptance rate: 17.2%)
10. *"Identifying Proxy Nodes in a Tor Anonymization Circuit"*  
Sambuddho Chakravarty, Angelos Stavrou, and Angelos D. Keromytis. In Proceedings of the

- 2<sup>nd</sup> Workshop on Security and Privacy in Telecommunications and Information Systems (SePTIS)*, pp. 633 - 639. December 2008, Bali, Indonesia. (Acceptance rate: 37.5%)
11. "*Online Network Forensics for Automatic Repair Validation*"  
Michael E. Locasto, Matthew Burnside, and Angelos D. Keromytis. In *Proceedings of the 3<sup>rd</sup> International Workshop on Security (IWSEC)*, pp. 136 - 151. November 2008, Kagawa, Japan. (Acceptance rate: 19.1%)
  12. "*Return Value Predictability for Self-Healing*"  
Michael E. Locasto, Angelos Stavrou, Gabriela F. Cretu, Angelos D. Keromytis, and Salvatore J. Stolfo. In *Proceedings of the 3<sup>rd</sup> International Workshop on Security (IWSEC)*, pp. 152 - 166. November 2008, Kagawa, Japan. (Acceptance rate: 19.1%)
  13. "*Asynchronous Policy Evaluation and Enforcement*"  
Matthew Burnside and Angelos D. Keromytis. In *Proceedings of the 2<sup>nd</sup> Computer Security Architecture Workshop (CSAW)*, pp. 45 - 50. October 2008, Fairfax, VA.
  14. "*Race to the bottom: Malicious Hardware*"  
Angelos D. Keromytis, Simha Sethumadhavan, and Ken Shepard. In *Proceedings of the 1<sup>st</sup> FORWARD Invitational Workshop for Identifying Emerging Threats in Information and Communication Technology Infrastructures*. April 2008, Goteborg, Sweden. (Invited paper)
  15. "*Arachne: Integrated Enterprise Security Management*"  
Matthew Burnside and Angelos D. Keromytis. In *Proceedings of the 8<sup>th</sup> Annual IEEE SMC Information Assurance Workshop (IAW)*, pp. 214 - 220. June 2007, West Point, NY.
  16. "*Poster Paper: Band-aid Patching*"  
Stelios Sidiroglou, Sotiris Ioannidis, and Angelos D. Keromytis. In *Proceedings of the 3<sup>rd</sup> Workshop on Hot Topics in System Dependability (HotDep)*, pp. 102 - 106. June 2007, Edinburgh, UK.
  17. "*Data Sanitization: Improving the Forensic Utility of Anomaly Detection Systems*"  
Gabriela F. Cretu, Angelos Stavrou, Salvatore J. Stolfo, and Angelos D. Keromytis. In *Proceedings of the 3<sup>rd</sup> Workshop on Hot Topics in System Dependability (HotDep)*, pp. 64 - 70. June 2007, Edinburgh, UK.
  18. "*Bridging the Network Reservation Gap Using Overlays*"  
Angelos Stavrou, David Michael Turner, Angelos D. Keromytis, and Vassilis Prevelakis. In *Proceedings of the 1<sup>st</sup> Workshop on Information Assurance for Middleware Communications (IAMCOM)*, pp. 1 - 6. January 2007, Bangalore, India.
  19. "*Next Generation Attacks on the Internet*"  
Evangelos Markatos and Angelos D. Keromytis. In *Proceedings (electronic) of the EU-US Summit Series on Cyber Trust: Workshop on System Dependability & Security*, pp. 67 - 73. November 2006, Dublin, Ireland. (Invited paper)
  20. "*Dark Application Communities*"  
Michael E. Locasto, Angelos Stavrou, and Angelos D. Keromytis. In *Proceedings of the New Security Paradigms Workshop (NSPW)*, pp. 11 - 18. September 2006, Schloss Dagstuhl, Germany.
  21. "*Privacy as an Operating System Service*"  
Sotiris Ioannidis, Stelios Sidiroglou, and Angelos D. Keromytis. In *Proceedings (electronic) of the 1<sup>st</sup> Workshop on Hot Topics in Security (HotSec)*. July 2006, Vancouver, Canada.
  22. "*PalProtect: A Collaborative Security Approach to Comment Spam*"  
Benny Wong, Michael E. Locasto, and Angelos D. Keromytis. In *Proceedings of the 7<sup>th</sup> Annual IEEE SMC Information Assurance Workshop (IAW)*, pp. 170 - 175. June 2006, West Point, NY.
  23. "*Adding a Flow-Oriented Paradigm to Commodity Operating Systems*"

- Christian Soviani, Stephen A. Edwards, and Angelos D. Keromytis. In Proceedings of the *Workshop on Interaction between Operating System and Computer Architecture (IOSCA)*, held in conjunction with the IEEE International Symposium on Workload Characterization, pp. 1 - 6. October 2005, Austin, TX.
24. "*Speculative Virtual Verification: Policy-Constrained Speculative Execution*"  
Michael E. Locasto, Stelios Sidiroglou, and Angelos D. Keromytis. In Proceedings of the *New Security Paradigms Workshop (NSPW)*, pp. 119 - 124. September 2005, Lake Arrowhead, CA.
  25. "*Application Communities: Using Monoculture for Dependability*"  
Michael E. Locasto, Stelios Sidiroglou, and Angelos D. Keromytis. In Proceedings of the *1<sup>st</sup> Workshop on Hot Topics in System Dependability (HotDep)*, held in conjunction with the International Conference on Dependable Systems and Networks (DSN), pp. 288 - 292. June 2005, Yokohama, Japan.
  26. "*Towards Collaborative Security and P2P Intrusion Detection*"  
Michael E. Locasto, Janak Parekh, Angelos D. Keromytis, and Salvatore J. Stolfo. In Proceedings of the *6<sup>th</sup> Annual IEEE SMC Information Assurance Workshop (IAW)*, pp. 333 - 339. June 2005, West Point, NY.
  27. "*FlowPuter: A Cluster Architecture Unifying Switch, Server and Storage Processing*"  
Alfred V. Aho, Angelos D. Keromytis, Vishal Misra, Jason Nieh, Kenneth A. Ross, and Yechiam Yemini. In Proceedings of the *1<sup>st</sup> International Workshop on Data Processing and Storage Networking: towards Grid Computing (DPSN)*, pp. 2/1 - 2/7. May 2004, Athens, Greece.
  28. "*One Class Support Vector Machines for Detecting Anomalous Windows Registry Accesses*"  
Katherine Heller, Krysta Svore, Angelos D. Keromytis, and Salvatore J. Stolfo. In Proceedings of the *ICDM Workshop on Data Mining for Computer Security*, held in conjunction with the *3<sup>rd</sup> International IEEE Conference on Data Mining*, pp. 2 - 9. November 2003, Melbourne, FL.
  29. "*A Holistic Approach to Service Survivability*"  
Angelos D. Keromytis, Janak Parekh, Philip N. Gross, Gail Kaiser, Vishal Misra, Jason Nieh, Dan Rubenstein, and Salvatore J. Stolfo. In Proceedings of the *1<sup>st</sup> ACM Workshop on Survivable and Self-Regenerative Systems (SSRS)*, held in conjunction with the *10<sup>th</sup> ACM International Conference on Computer and Communications Security (CCS)*, pp. 11 - 22. October 2003, Fairfax, VA.
  30. "*High-Speed I/O: The Operating System As A Signalling Mechanism*"  
Matthew Burnside and Angelos D. Keromytis. In Proceedings of the *ACM SIGCOMM Workshop on Network-I/O Convergence: Experience, Lessons, Implications (NICELI)*, held in conjunction with the *ACM SIGCOMM Conference*, pp. 220 - 227. August 2003, Karlsruhe, Germany.
  31. "*A Network Worm Vaccine Architecture*"  
Stelios Sidiroglou and Angelos D. Keromytis. In Proceedings of the *12<sup>th</sup> IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), Workshop on Enterprise Security*, pp. 220 - 225. June 2003, Linz, Austria.
  32. "*Design and Implementation of Virtual Private Services*"  
Sotiris Ioannidis, Steven M. Bellovin, John Ioannidis, Angelos D. Keromytis, and Jonathan M. Smith. In Proceedings of the *12<sup>th</sup> IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), Workshop on Enterprise Security, Special Session on Trust Management in Collaborative Global Computing*, pp. 269 - 274. June 2003, Linz, Austria.



33. "*WebDAVA: An Administrator-Free Approach To Web File-Sharing*"  
Alexander Levine, Vassilis Prevelakis, John Ioannidis, Sotiris Ioannidis, and Angelos D. Keromytis. In Proceedings of the 12<sup>th</sup> IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE), Workshop on Distributed and Mobile Collaboration, pp. 59 - 64. June 2003, Linz, Austria.
34. "*Protocols for Anonymity in Wireless Networks*"  
Matt Blaze, John Ioannidis, Angelos D. Keromytis, Tal Malkin, and Avi Rubin. In Proceedings of the 11<sup>th</sup> International Workshop on Security Protocols. April 2003, Cambridge, England.
35. "*xPF: Packet Filtering for Low-Cost Network Monitoring*"  
Sotiris Ioannidis, Kostas G. Anagnostakis, John Ioannidis, and Angelos D. Keromytis. In Proceedings of the Workshop on High Performance Switching and Routing (HPSR), pp. 121 - 126. May 2002, Kobe, Japan.
36. "*Toward Understanding the Limits of DDoS Defenses*"  
Matt Blaze, John Ioannidis, and Angelos D. Keromytis. In Proceedings of the 10<sup>th</sup> International Workshop on Security Protocols, Springer-Verlag Lecture Notes in Computer Science, vol. 2467. April 2002, Cambridge, England.
37. "*Toward A Unified View of Intrusion Detection and Security Policy*"  
Matt Blaze, Angelos D. Keromytis, and Salvatore J. Stolfo. In Proceedings of the 10<sup>th</sup> International Workshop on Security Protocols, Springer-Verlag Lecture Notes in Computer Science, vol. 2467. April 2002, Cambridge, England.
38. "*Efficient, DoS-resistant, Secure Key Exchange for Internet Protocols*"  
William Aiello, Steven M. Bellovin, Matt Blaze, Ran Canetti, John Ioannidis, Angelos D. Keromytis, and Omer Reingold. In Proceedings of the 9<sup>th</sup> International Workshop on Security Protocols, Springer-Verlag Lecture Notes in Computer Science, vol. 2133, pp. 40 - 48. April 2001, Cambridge, England.
39. "*Scalable Resource Control in Active Networks*"  
Kostas G. Anagnostakis, Michael W. Hicks, Sotiris Ioannidis, Angelos D. Keromytis, and Jonathan M. Smith. In Proceedings of the 2<sup>nd</sup> International Workshop for Active Networks (IWAN), pp. 343 - 357. October 2000, Tokyo, Japan.
40. "*A Secure Plan*"  
Michael W. Hicks and Angelos D. Keromytis. In Proceedings of the 1<sup>st</sup> International Workshop for Active Networks (IWAN), pp. 307 - 314. June - July 1999, Berlin, Germany. An extended version is available as *University of Pennsylvania Technical Report MS-CIS-99-14*, and was also published in the Proceedings of the *DARPA Active Networks Conference and Exposition (DANCE)*, May 2002.
41. "*Trust Management and Network Layer Security Protocols*"  
Matt Blaze, John Ioannidis, and Angelos D. Keromytis. In Proceedings of the 7<sup>th</sup> International Workshop on Security Protocols, Springer-Verlag Lecture Notes in Computer Science, vol. 1796, pp. 103 - 108. April 1999, Cambridge, England.
42. "*The SwitchWare Active Network Implementation*"  
D. Scott Alexander, Michael W. Hicks, Pankaj Kakkar, Angelos D. Keromytis, Marianne Shaw, Jonathan T. Moore, Carl A. Gunter, Trevor Jim, Scott M. Nettles, and Jonathan M. Smith. In Proceedings of the *ACM SIGPLAN Workshop on ML*, held in conjunction with the *International Conference on Functional Programming (ICFP)*, pp. 67 - 76. September 1998, Baltimore, MD.
43. "*KeyNote: Trust Management for Public-Key Infrastructures*"  
Matt Blaze, Joan Feigenbaum, and Angelos D. Keromytis. In Proceedings of the 6<sup>th</sup>

*International Workshop on Security Protocols*, Springer-Verlag Lecture Notes in Computer Science, vol. 1550, pp. 59 - 63. April 1998, Cambridge, England. Also available as *AT&T Technical Report 98.11.1*.

#### **Additional Publications**

1. *"Transport Layer Security (TLS) Authorization Using KeyNote"*  
Angelos D. Keromytis. *Request For Comments (RFC) 6042*, October 2010.
2. *"X.509 Key and Signature Encoding for the KeyNote Trust Management System"*  
Angelos D. Keromytis. *Request For Comments (RFC) 5708*, January 2010.
3. *"SSARES: Secure Searchable Automated Remote Email Storage"*  
Adam J. Aviv, Michael E. Locasto, Shaya Potter, and Angelos D. Keromytis. In the Columbia Computer Science Student Research Symposium, Fall 2006.
4. *"IP Security Policy Requirements"*  
Matt Blaze, Angelos D. Keromytis, Michael Richardson, and Luis Sanchez. *Request For Comments (RFC) 3586*, August 2003.
5. *"On the Use of Stream Control Transmission Protocol (SCTP) with IPsec"*  
Steven M. Bellovin, John Ioannidis, Angelos D. Keromytis, and Randal R. Stewart. *Request For Comments (RFC) 3554*, June 2003.
6. *"The Use of HMAC-RIPEDM-160-96 within ESP and AH"*  
Angelos D. Keromytis and Niels Provos. *Request For Comments (RFC) 2857*, June 2000.
7. *"DSA and RSA Key and Signature Encoding for the KeyNote Trust Management System"*  
Matt Blaze, John Ioannidis, and Angelos D. Keromytis. *Request For Comments (RFC) 2792*, March 2000.
8. *"The KeyNote Trust-Management System, Version 2"*  
Matt Blaze, Joan Feigenbaum, John Ioannidis, and Angelos D. Keromytis. *Request For Comments (RFC) 2704*, September 1999.

#### **Technical Reports/Works in Progress**

1. *"Symantec Report on Rogue Security Software, July 2008 - June 2009"*  
Marc Fossi, Dean Turner, Eric Johnson, Trevor Mack, Teo Adams, Joseph Blackbird, Mo King Low, David McKinney, Marc Dacier, Angelos D. Keromytis, Corrado Leita, Marco Cova, Jon Orbeton, and Olivier Thonnard. Symantec Technical Report, October 2009.
2. *"LinkWidth: A Method to Measure Link Capacity and Available Bandwidth using Single-End Probes"*  
Sambuddho Chakravarty, Angelos Stavrou, and Angelos D. Keromytis. *Columbia University Computer Science Department Technical Report CUCS-002-08*, January 2008.
3. *"Can P2P Replace Direct Download for Content Distribution?"*  
Alex Sherman, Angelos Stavrou, Jason Nieh, Cliff Stein, and Angelos D. Keromytis. *Columbia University Computer Science Department Technical Report CUCS-020-07*, March 2007.
4. *"A Model for Automatically Repairing Execution Integrity"*  
Michael E. Locasto, Gabriela F. Cretu, Angelos Stavrou, and Angelos D. Keromytis. *Columbia University Computer Science Department Technical Report CUCS-005-07*, January 2007.
5. *"Speculative Execution as an Operating System Service"*  
Michael E. Locasto and Angelos D. Keromytis. *Columbia University Computer Science Department Technical Report CUCS-024-06*, May 2006.
6. *"Quantifying Application Behavior Space for Detection and Self-Healing"*

- Michael E. Locasto, Angelos Stavrou, Gabriela F. Cretu, Angelos D. Keromytis, and Salvatore J. Stolfo. *Columbia University Computer Science Department Technical Report CUCS-017-06*, April 2006.
7. "*Bloodhound: Searching Out Malicious Input in Network Flows for Automatic Repair Validation*"  
Michael E. Locasto, Matthew Burnside, and Angelos D. Keromytis. *Columbia University Computer Science Department Technical Report CUCS-016-06*, April 2006.
  8. "*Binary-level Function Profiling for Intrusion Detection and Smart Error Virtualization*"  
Michael E. Locasto and Angelos D. Keromytis. *Columbia University Computer Science Department Technical Report CUCS-002-06*, January 2006.
  9. "*A General Analysis of the Security of Elastic Block Ciphers*"  
Debra Cook, Moti Yung, and Angelos D. Keromytis. *Columbia University Computer Science Department Technical Report CUCS-038-05*, September 2005.
  10. "*The Pseudorandomness of Elastic Block Ciphers*"  
Debra Cook, Moti Yung, and Angelos D. Keromytis. *Columbia University Computer Science Department Technical Report CUCS-037-05*, September 2005.
  11. "*PachyRand: SQL Randomization for the PostgreSQL JDBC Driver*"  
Michael E. Locasto and Angelos D. Keromytis. *Columbia University Computer Science Department Technical Report CUCS-033-05*, August 2005.
  12. "*Elastic Block Ciphers: The Feistel Cipher Case*"  
Debra L. Cook, Moti Yung, and Angelos D. Keromytis. *Columbia University Computer Science Department Technical Report CUCS-021-04*, May 2004.
  13. "*Collaborative Distributed Intrusion Detection*"  
Michael E. Locasto, Janak J. Parekh, Salvatore J. Stolfo, Angelos D. Keromytis, Tal Malkin, and Vishal Misra. *Columbia University Computer Science Department Technical Report CUCS-012-04*, March 2004.
  14. "*Elastic Block Ciphers*"  
Debra L. Cook, Moti Yung, and Angelos D. Keromytis. *Columbia University Computer Science Department Technical Report CUCS-010-04*, February 2004.
  15. "*Just Fast Keying (JFK)*"  
William Aiello, Steven M. Bellovin, Matt Blaze, Ran Canetti, John Ioannidis, Angelos D. Keromytis, and Omer Reingold. *IETF IPsec Working Group*, April 2002,.
  16. "*CASPER: Compiler-Assisted Securing of Programs at Runtime*"  
Gaurav S. Kc, Stephen A. Edwards, Gail E. Kaiser, and Angelos D. Keromytis. *Columbia University Computer Science Department Technical Report CUCS-025-02*, 2002.
  17. "*The 'suggested ID' extension for IKE*"  
Angelos D. Keromytis and William Sommerfeld. *IETF IPsec Working Group*, November 2001.
  18. "*SPKI: ShrinkWrap*"  
Angelos D. Keromytis and William A. Simpson. *IETF SPKI Working Group*, September 1997.
  19. "*Active Network Encapsulation Protocol (ANEP)*"  
D. Scott Alexander, Bob Braden, Carl A. Gunter, Alden W. Jackson, Angelos D. Keromytis, Gary J. Minden, and David Wetherall. *Active Networks Group, DARPA Active Networks Project*, August 1997.

20. *"Creating Efficient Fail-Stop Cryptographic Protocols"*  
Angelos D. Keromytis and Jonathan M. Smith. *University of Pennsylvania Technical Report MS-CIS-96-32*, December 1996.

DM\_US 34918171-2.077580.0132

# EX. 1057

For Declaration of Chris Hopen



## HomePipe Networks, Inc.

[About](#) | [Press Room](#) | [Personnel](#) | [Contact](#)

### Chris Hopen CEO and Co-Founder

“ Hopen is a recognized pioneer in the SSL-VPN market and technology space. An experienced engineering leader and technologist in networking and security, Hopen was the co-founder and CTO for Aventail Corporation, later acquired by SonicWall (SNWL). ”

Chris Hopen brings more than 20 years of experience as an engineering leader and technologist in networking and security. Prior to HomePipe, Chris was the co-founder and chief technical officer for Aventail Corporation, which was acquired by SonicWall (SNWL). Recognized as a leader, Chris pioneered the SSL-VPN market and technology solution space. His accomplishments include network security-related patents and Internet Engineering Task Force (IETF) publications as well as being named one of the top 50 IT executives in the service provider sector by InfoWorld. Chris holds a B.S. in Computer Science, Mathematics and Economics from Western Washington University and actively works with the W.W.U School of Business and Computer Science Advisory Board.

Chris Hopen  
CEO and Co-Founder  
HomePipe Networks, Inc.

**Website:**  
[www.homepipe.net](http://www.homepipe.net)

**Address:**  
Seattle, Washington  
United States

**Areas of Expertise:**  
Networking  
Security

---

**Disclaimer:** Users are solely responsible for the content posted by them. PRLog can't be held liable for the content posted by others. [Report Abuse](#)

# EX. 1058

For Declaration of Chris Hopen



Options

Hello, jeff. | [Your account](#) | [Help](#) | [Log out](#)

Browse by publication

Follow us:

[Home](#) » [Publications](#) » [U.S. newspapers and newswires](#) » [U.S. newswires](#) » [PR Newswire](#) » [Apr - Jun 1997](#) » [May 2, 1997](#)



# Aventail Ships the First Standards-Based Virtual Private Network Software Solution

Publication: **PR Newswire** Publish date: **May 2, 1997**

Like 0 Share 0

## Aventail MobileVPN and PartnerVPN Include Granular Access Controls and Support

For Multiple Authentication and Encryption Methods

SEATTLE, May 2 /PRNewswire/ -- Aventail Corporation announced today the availability of the industry's only standards-based Virtual Private Network (VPN) software solutions. Aventail MobileVPN and Aventail PartnerVPN for Windows NT will begin shipping today and pricing starts at \$4,995. UNIX versions will be available at the end of this month.

Aventail MobileVPN and Aventail PartnerVPN enable organizations to securely communicate over the Internet, allowing companies to extend the reach of their corporate intranet to customers, partners, remote offices, and mobile employees. Aventail's adherence to standards simplifies VPN deployment, enables interoperability, and leverages corporations' existing network investments.

"Aventail has moved the concept of a VPN to the next level. They are the only company providing a highly secure circuit-level solution that is deployable over existing network infrastructure and has the ability to work with a variety of authentication and encryption technologies," says Ira Machevsky, vice president at Giga Information Group.

### The Only Standards-Based VPN Product

Aventail MobileVPN and Aventail PartnerVPN are the first VPN solutions based on SOCKS v5, an open Internet Engineering Task Force (IETF) standard. SOCKS is a distributed network security standard that represents the next-generation of Internet security. The SOCKS protocol has received widespread support from leading Internet vendors, including Netscape (Nasdaq: NSCP), Microsoft (Nasdaq: MSFT), IBM (NYSE: IBM), Sterling Software (NYSE: SSW), NetManage (Nasdaq: NETM), FTP Software (Nasdaq: FTPS), and Pointcast. NEC USA, Incorporated has been the driving force behind SOCKS with the vision that it would be the most important communication technology for the Internet.

### Powerful Security and Management Tools

Aventail MobileVPN and Aventail PartnerVPN are the only products to support all of the popular authentication and encryption methods, such as SSL, DES, TripleDES, CHAP, RC4, MD4, MD5, and RADIUS. Other features include:

- \* Access Control Tool allows the IS administrator to specify access based on destination, source, application usage, type of encryption and/or authentication, and specific filtering profiles.
- \* Protocol Filtering blocks specific JAVA, ActiveX or any other application that could demand too much bandwidth or infect the network with a virus.
- \* Content Filtering blocks out objectionable content that may interfere with employee productivity.
- \* Traffic Monitor shows real-time inbound and outbound traffic through a graphical interface.
- \* Reporting and Logging Tool monitors and logs server activity so that reports can easily be produced from any SQL supported database.
- \* Administration Tool enables IS managers to easily configure the server and add or modify security or management modules.

### Product Demonstrations at Network+Interop

Aventail will be conducting product demonstrations in Booth 1710 at Network+Interop in Las Vegas from May 6th to 8th.

### About Aventail

Aventail Corporation is the leading developer of Virtual Private Network (VPN) solutions. Aventail software allows organizations to build session-layer VPNs so corporations can privately communicate with mobile employees, remote offices, and business partners. Aventail's standards-based products represent the next-level of secure communication by providing strong authentication and encryption, customizable access controls, comprehensive monitoring, logging and reporting capabilities.

Aventail offers four security solutions: Aventail MobileVPN, Aventail PartnerVPN, Aventail Internet

#### Article tools

[Save this article](#)

[Print this article](#)

[E-mail this article](#)

[Export to Microsoft Word](#)

[Cite this article](#)

[Related articles](#)

#### Research Center

[All saved items](#)

[Saved searches](#)

[Saved articles](#)

[Alerts](#)

[Your account](#)

Actually, the world DOES need another lawyer.

**DiscoverLaw.org**

**HighBeam Research on Facebook**

Like

684 people like HighBeam Research.

Facebook social plugin

**Want help with tests and projects?**

Get study tools specific to your textbook!

- Printed texts
- Lab manuals
- Solutions manuals
- Study guides
- eBooks
- Single eChapters

CENGAGE brain Find your textbook



Policy Manager (IPM), and Aventail AutoSOCKS. Aventail MobileVPN enables mobile or remote employees to have secure and managed access into the corporate network. Aventail PartnerVPN allows a company to extend their network to customers, suppliers, remote offices or corporate partners. Aventail IPM allows corporations to control and implement their Internet security policies. Aventail AutoSOCKS enables client TCP/IP applications to securely traverse existing SOCKS-based firewalls and servers.

The company has offices in Seattle, Washington and can be contacted by phone: 888-SOCKSV5 (762-5785), fax: 206-777-5656, or email: info@aventail.com. Aventail's Web address is www.aventail.com.

NOTE: Aventail, MobileVPN, and PartnerVPN are trademarks of Aventail Corporation. All other brands, products, and service names mentioned are trademarks or registered service marks of their respective owners.

SOURCE Aventail Corporation

-0- 05/02/97

/CONTACT: Deanna Leung of Aventail Corporation, 206-777-5617, or deanna@aventail.com; or Jessica Maco of Reed, Revell-Pechar, Inc., 206-462-4777, or jmaco@rrp.com/

CO: Aventail Corporation ST: Washington IN: CPR MLM SU: PDT

DC-KW -- SFF006 -- 9928 05/02/97 08:01 EDT http://www.prnewswire.com

COPYRIGHT 2009 PR Newswire Association LLC. This material is published under license from the publisher through the Gale Group, Farmington Hills, Michigan. All inquiries regarding rights should be directed to the Gale Group. For permission to reuse this article, contact [Copyright Clearance Center](#).




**Cite this article**

Pick a style below, and copy the text for your bibliography.

MLA Chicago APA [Learn more about citation styles](#)

"Aventail Ships the First Standards-Based Virtual Private Network Software Solution." PR Newswire. PR Newswire Association LLC. 1997. *HighBeam Research*. 13 May. 2011 <<http://www.highbeam.com>>.

**More articles like this:**

-  [Aventail Moves Beyond Insecure Tunnels, Rolls Out the Industry's First ...](#)  
PR Newswire; March 10, 1997 ; 700+ words ... TM) and **Aventail PartnerVPN**(TM) Combine ... security risk. **Aventail Corporation** announced ... TM) and **Aventail PartnerVPN**(TM) ... and CEO of **Aventail Corporation**. "Unlike ... MobileVPN and ...
-  [Aventail Announces the First VPN Solution to Assure Interoperability ...](#)  
PR Newswire; June 2, 1997 ; 700+ words ... parameters. About Aventail **Aventail Corporation** is the leading developer ... solutions: Aventail MobileVPN, **Aventail PartnerVPN**, Aventail Internet Policy ... aventail.com. SOURCE **Aventail Corporation** -0- 06/02/97 /CONTACT ...
-  [NetManage is the First PC Connectivity Vendor to Embrace Socks v5 ...](#)  
PR Newswire; April 28, 1997 ; 700+ words ... PRNewswire/ -- **Aventail Corporation** today announced ... president & CEO of **Aventail Corporation**. "NetManage ... About Aventail **Aventail Corporation** is the leading ... Aventail MobileVPN, **Aventail** ...

[See all results](#)

**Find articles, research, and archives**

[HighBeam® Research](#), a part of The Gale Group, Inc. © Copyright 2011. All rights reserved.  
[Home](#) [About us](#) [Customer support](#) [Group subscriptions](#) [Advertising](#) [Partnerships](#) [Privacy policy](#) [Terms and conditions](#)

The HighBeam advertising network includes: [womensforutps.com](#) [Glam](#) [Family](#)

# EX. 1059

For Declaration of Chris Hopen



2 of 2 DOCUMENTS

Copyright 1997 InfoWorld Media Group  
InfoWorld

June 23, 1997

**SECTION:** NETWORKING: Product Reviews; Pg. 64d

**LENGTH:** 1067 words

**HEADLINE:** Aventail delivers highly secure, flexible VPN solution

**BYLINE:** By Lai-Han Szeto

**BODY:**

For secure remote-access needs, Aventail's MobileVPN 2.0 and AutoSocks 2.1 comprise a virtual private network (VPN) software solution that lets you monitor and maintain access to your central site via application-level proxies.

Most VPN products, such as Microsoft's Steelhead technology, Digital's AltaVista Tunnel, and Data Fellow's F-Secure, do not address security issues beyond initial log-ins, tending to be server-centric. Aventail has engineered a solution that is user-centric, taking a more in-depth approach to VPN implementation.

Boasting nearly unmatched interoperability with other security protocols, MobileVPN and AutoSocks succeed as a VPN solution, but not without drawbacks: Unidirectional data flow prohibits broadcasting and remote administration, and the system requires third-party products for specific IP-layer features, such as IPX encapsulation.

High level of security

Aventail has developed its own connectivity protocol, Socks 5, which represents the next step in the evolution of the well-known Socks 4 protocol. The addition of security protocols makes Socks 5 a viable VPN tool and a contender to Microsoft's Point to Point Tunneling Protocol (PPTP). Aventail implements the Socks 5 protocol in the Aventail Server, the engine of its VPN package. Socks 5 is based on directed architecture, as opposed to the tunneled architecture one usually associates with VPN technology.

The server establishes a unidirectional connection with a remote client (AutoSocks) or second host site. A secured user can read, write, and execute to the host Server site according to the user's permission profile, but the host cannot likewise carry out transactions on the user's machine. This

setup prevents an intruder from accessing both sites.

Unlike IP-based protocols such as IP Security Architecture (IPSec), a tunneling protocol currently in the draft stage, Socks 5 compels a user to pass permission requirements once that user passes the system perimeter. Once users traverse firewalls, Socks 5 limits access to specific parts of your host system. The system locks out users from directories and applications according to their permission profile.

Socks 5 performs encryption and authentication at the session layer (Layer 5) of the IP packet, enabling an interoperability unmatched by most of Aventail's competitors.

Aventail products support Challenge Handshake Authentication Protocol, Secure Sockets Layer, and Remote Access Dial-In User Service authentication. In addition, Aventail deploys an open architecture to further enhance the flexibility of its products. Key management is compliant with Public Key Cryptography Standards. Encryption is DES and triple-DES enabled. Recently, Aventail announced Socks 5 capability with the IPSec, PPTP, and Layer 2 Tunneling Protocol security protocols.

#### Outside authority

MobileVPN represents an achievement in usability. I ran my VPN server on Windows NT 4.0 and used a Windows 95 client unit running AutoSocks.

MobileVPN carries handy administrative tools such as Proxy Chaining and Credential Caching, as well as myriad conventional utilities for alias tables, filtering, and session parameters.

AutoSocks acts as the remote-access agent that intercepts application requests between the client application itself and the WinSock interface. It offers logging and configuration GUIs that resemble a miniature version of MobileVPN, minus the high-level host controls.

I installed both pieces with minimal hassle, minus a certificate authority component. Aventail has no plans to become a certificate authority vendor, leaving the task to third parties, such as VeriSign. Unfortunately, this extra service can cost from \$290 to as much as \$2,000 per year per server.

Add this to Aventail's tiered licensing scheme, and the bottom line becomes a little steeper than that of most conventional VPN solutions. Whether it is worth the cost depends on the complexity of your security policies.

#### Fluctuating protocols

Implementing VPNs is not for the faint of heart or pocketbook. Tunneling protocols are maturing even as I write. The key to maintaining a foothold in the market is flexibility. In general, developers are building modular products in anticipation of the Internet Engineering Task Force's final draft of IPSec. It is hard to say what will become of Socks 5 (or Socks 6), but for now it has found a little-explored niche in secured connectivity.

Aventail delivers highly secure, flexible VPN solution InfoWorld June 23, 1997

Although MobileVPN and AutoSocks lack bidirectional communication and IP-layer features, their open architecture makes them compatible with multiple standards and provides a high level of security.

Lai-Han Szeto (laihan\_szeto@infoworld.com) is a contract analyst at the InfoWorld Test Center.

**THE BOTTOM LINE: EXCELLENT**

MobileVPN 2.0 and AutoSocks 2.1

This virtual private network (VPN) software combination offers a secure and easy-to-manage remote-access solution.

**Pros:** Excellent proxy-level management; flexible architecture that complements other VPN and security products.

**Cons:** Third-party products required for specific IP-layer features such as IPX encapsulation; no broadcasting or remote administration.

Aventail Corp., Seattle; (888) 762-5785 (toll-free), (206) 777-5600; fax: (206) 777-5656; <http://www.aventail.com>.

**Price:** \$4,999 per server for fewer than 25 connections; \$66 per client seat for fewer than 25 seats. (Tiered pricing available.)

**Platforms:** MobileVPN: Unix, Windows NT; AutoSocks: Unix, Windows 3.x, Windows 95, Windows NT.

**LOAD-DATE:** June 23, 1997

# EX. 1060

For Declaration of Chris Hopen



## **Aventail Introduces The First Extranet-Ready Platform; Aventail Previews its Latest Solution, Aventail ExtraNet Center, at Network+Interop in Atlanta.**

Publication: **PR Newswire** Publish date: **October 12, 1998**

SEATTLE, Oct. 12 /PRNewswire/ -- With a strong reputation for providing easy-to-manage security software solutions, Aventail Corporation today announced the introduction of Aventail ExtraNet Center(TM).

Aventail ExtraNet Center enables corporations to securely extend their enterprise applications to business partners, suppliers, and customers over the Internet and other public networks. It is the only extranet-ready platform that delivers the necessary security, centralized management, and application and network integration for building an extranet, eliminating the barriers that have previously deterred the widespread deployment of extranets.

"Businesses are realizing that in order to stay competitive in this global market, they must constantly evaluate and improve their business processes. It is imperative that they think of innovative ways to improve the delivery of information to key individuals," said Evan Kaplan, president & CEO of Aventail Corporation. "Aventail ExtraNet Center enables corporations to improve customer relations, facilitate collaborative projects with partners, and increase employee productivity."

### **Today's Extranet Challenges Addressed**

Aventail Extranet Center is a client/server software solution that addresses the specific security, management, and deployment issues that many corporations face when designing a system to share information with extranet users.

The complete solution provides the following:

- \* **Sophisticated Security and Access Controls** Aventail ExtraNet Center not only provides strong encryption and authentication, but also granular access controls that enable administrators to define user privileges based on a broad range of parameters, including authentication/encryption method, user ID, information resource, group affiliation, and day and time. This provides corporations with the flexibility to build custom access profiles to reflect the unique business relationship of each partner.

- \* **Application Independent**

Currently, many corporations are deploying extranets with products that only support Web-based applications, greatly limiting the functionality of the extranet. Aventail ExtraNet Center supports all IP-based applications including legacy host, Web, JAVA, ActiveX, CORBA, DCOM+, custom corporate, and client/server applications from corporations such as SAP, BAAN and PeopleSoft.

- \* **Simple User Management**

Aventail ExtraNet Center makes deployment easy for the administrator, even if there are thousands or millions of users. Through the Aventail Policy Console, a single intuitive interface, administrators can easily create, delete, or modify extranet users' profiles. Using the Aventail Management Console, these functionalities can also be securely administered from any remote or desktop workstation.

- \* **Infrastructure Independent**

Aventail ExtraNet Center runs on most operating systems and works with any firewall, encryption and authentication method, and proxy server. The ability to seamlessly integrate into any existing infrastructure allows corporations to leverage their existing and future network and security infrastructure investments. In addition, it makes it easy for corporations and their business partners to select "best-of-breed" technologies that address their specific business requirements.

- \* **Transparent Client**

Designed for non-technical users, Aventail Extranet Client is a highly functional piece of software that is completely transparent to the end user. It can be installed in minutes, makes no technical modifications to the desktop, and can securely traverse any firewall without administrator intervention. Aventail Extranet Client includes Extranet Neighborhood, a revolutionary application that enables users to browse selected 32-bit

Windows-based file systems. Using the popular and well-known Microsoft Windows Explorer user interface, Extranet Neighborhood does not require any end-user training.

#### \* Automated Client Configuration and Distribution

Network administrators can create up to tens of thousands of custom Aventail Extranet Clients in one easy step with the Aventail Customizer. With this tool, network administrators can easily distribute clients and make them available in a central, networked directory for easy access, download, and installation.

"Information Systems (IS) decision makers charged with protecting core business information and the brand-equity of the enterprise will want to look at Aventail ExtraNet Center," said Jim Hurlley, managing director of the information security practice with Aberdeen Group. "Our clients are wrestling with how to safely deploy and maintain enterprise commerce activities with key customers, partners and suppliers. Aventail ExtraNet Center is positioned to meet this need."

#### Real-Life Extranet Deployments

To date, many organizations in the healthcare, financial services, consulting, manufacturing, insurance, and high technology industries have benefited greatly from successful extranet deployments using Aventail solutions.

For example, Ari Friedman, a network engineer at University Health Systems, uses Aventail to provide doctors, medical students, and staff members at The University of Texas Health Science Center in San Antonio with secure access to patient billing, scheduling, and lab results. By deploying Aventail's extranet solution, the users at the Health Science Center can be more productive and spend more time with their patients, giving them better care.

"I evaluated several VPN hardware products and firewall solutions. The hardware products were unacceptable because the client would have been a nightmare to deploy, and it meant managing and purchasing another device. Firewalls were out of the question because they provided terrible performance," said Friedman. "Aventail meets all of my requirements. Their solution is dependable, versatile, and something that scales well."

#### Pricing and Availability

Aventail ExtraNet Center will be available in November through Aventail's worldwide sales team and Aventail's Extranet Advantage VAR partners. Pricing details will be available at the time of product release.

#### About Aventail

Aventail is at the forefront of providing extranet security and management software solutions that allow organizations to securely extend their enterprise applications to strategic partners, suppliers, customers, consultants, and other key individuals. Aventail's extranet solutions enable organizations to increase their competitive advantage, raise profits, and leverage their investments in existing and future enterprise systems. With a strong reputation for providing highly secure and easy-to-manage software solutions, Aventail has received numerous industry awards from publications such as InfoWorld, Network Computing, LAN Times, BYTE Magazine, Software Digest, and Computer Reseller News.

Aventail Corporation is a privately held company headquartered in Seattle, Washington. For more information on the company and its products, please visit the company's Web site at [www.aventail.com](http://www.aventail.com), or contact the company directly at 206-215-1111, 877-AVENTAIL, or [info@aventail.com](mailto:info@aventail.com).

Aventail and Aventail ExtraNet Center are trademarks of Aventail Corporation. All other trademarks are the property of their respective owners.

COPYRIGHT 2009 PR Newswire Association LLC. This material is published under license from the publisher through the Gale Group, Farmington Hills, Michigan. All inquiries regarding rights should be directed to the Gale Group. For permission to reuse this article, contact [Copyright Clearance Center](http://www.copyright.com).

HighBeam® Research, a part of The Gale Group, Inc. © Copyright 2011. All rights reserved. [www.highbeam.com](http://www.highbeam.com)

The HighBeam advertising network includes: [womenforutms.com](http://www.womenforutms.com) [iamfamily.com](http://www.iamfamily.com)



# EX. 1061

For Declaration of Chris Hopen



1 of 8 DOCUMENTS

Copyright 1998 Network World, Inc.  
Network World

October 19, 1998

**SECTION:** INTRANET APPS; Pg. 55

**LENGTH:** 250 words

**HEADLINE:** Briefs

**BODY:**

Aventail Corp. last week introduced the Aventail ExtraNet Center 3.0. This client/server package provides access controls, user-based authentication and key-certificate management and active filtering for business partners and suppliers who communicate over the Internet. The Aventail ExtraNet Center, which starts at \$7,995, is available for Windows NT 4.0, Linux 2.X, and Unix platforms from Digital, Sun and Hewlett-Packard. Aventail: (206) 215-1111

=09 John Manley, Canadian Minister of Industry, recently announced the government of Canada wants to make it easier to export products with encryption features in order to encourage electronic commerce. Canada will streamline export procedures with a one-time review process for even the strongest encryption, without requiring key-recovery features. More information is available at the Canadian government's Web site at <http://info.ic.gc.ca/cmb/welcomeic.nsf/Pages/releasefr.htm>.

IBM has released the beta of its HotMedia Web multimedia toolkit, a set of Java applets and assembly tools that let Web developers add sound and video clips to Web presentations. IBM is integrating HotMedia into the IBM electronic commerce server, net.commerce. Several other electronic catalog companies, including iCat, InterShop and Open Market, are beta-testing HotMedia with an eye toward the same goal. Now available for free download at [w.software.ibm.com/netmedia](http://w.software.ibm.com/netmedia), HotMedia will have a licensing fee when it formally ships by year-end.

**LOAD-DATE:** November 6, 1998