UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

APPLE INC.,

Petitioner,

v.

VirnetX, Inc. and Science Application International Corp.,

Patent Owner.

Patent No. 8,868,705
Issued: October 21, 2014
Filed: September 13, 2012
Inventors: Victor Larson, *et al.*
Title: AGILE NETWORK PROTOCOL FOR SECURE COMMUNICATIONS USING SECURE DOMAIN NAMES

Patent No. 8,850,009
Issued: September 30, 2014
Filed: June 6, 2013
Inventors: Victor Larson, *et al.*
Title: SYSTEM AND METHOD EMPLOYING AN AGILE NETWORK PROTOCOL FOR SECURE COMMUNICATIONS USING SECURE DOMAIN NAMES

Inter Partes Review Nos. IPR2015-00810, IPR2015-00811, IPR2015-00812 and IPR2015-00813

DECLARATION OF ROBERTO TAMASSIA REGARDING U.S. PATENT NOS. 8,868,705 AND 8,850,009

TABLE OF CONTENTS

I. INTRODUCTION 1

A. Engagement 1

B. Background and Qualifications 1

C. Compensation and Prior Testimony 3

D. Information Considered 3

II. LEGAL STANDARDS FOR PATENTABILITY 4

A. Anticipation 6

B. Obviousness 7

III. THE '705 AND '009 PATENTS 12

A. Effective Filing Dates 13

1. Effective Filing Date of the '705 Patent 13

2. Effective Filing Date of the '009 Patent 15

B. Overview of The '705 and '009 Patents 17

C. The Prosecution History of The '705 and '009 Patents 20

1. The '705 Patent 20

2. The '009 Patent 21

D. Construction of Terms Used in the '705 and '009 Patent Claims 22

1. Background on the Broadest Reasonable Interpretation 22

2. Broadest Reasonable Interpretation of Terms of the '705 Patent 24

a) "intercepting . . . a request" 24

b) "domain name" 25

c) "secure domain name" 26

d) "provisioning information" 27

e) "modulated transmission link" / "unmodulated transmission link" 29

f) "phone" 31
3. Broadest Reasonable Interpretation of Terms of the '009 Patent 32
a) "domain name service (DNS) request" 32
b) "interception of the DNS request" 33
c) "encrypted communication link" 33
d) "provisioning information" 35
e) "secure communications service" 35
f) "indication" 36
g) "virtual private network communication link" 38
h) "domain name" 39
i) "modulation" 40
E. Level of Ordinary Skill in the Art 40

IV. TECHNICAL BACKGROUND 41

A. COMPUTER NETWORKS 41
j) The OSI Model 41
k) The Internet 44
B. Internet Protocol Suite (TCP/IP) 45
C. DOMAIN NAME SYSTEM (DNS) 47
D. NETWORK ENCRYPTION 50
1. Symmetric Encryption – DES & AES 51
2. Asymmetric Key Encryption – Public/Private Keys 52
3. Key Exchange 53

V. IDENTIFICATION OF THE PRIOR ART 56

A. Exhibits 1009-1011 – The Aventail References 56
B. Exhibit 1007 – U.S. Patent 6,496,867 to Beser and Borella 57
C. Request for Comment (RFC) Publications 58
D. Exhibit 1008 – RFC 2401 60
E. Exhibit 1013 – RFC 2543 60

VI. OVERVIEW OF THE PRIOR ART 61

A. Aventail 61

1. Overview of Aventail Extranet Center 63
2. Aventail Connect.....66
3. Aventail Extranet Server.....67
4. Aventail Extranet VPN 71
5. Types of Communication Supported by Aventail 76
6. Physical Implementation of the Networks Used by Aventail.....77
7. Security and Encryption in Aventail.....78
   a) SOCKS v5 79
   b) Authentication Modules 81
8. Aventail Extranet Center – Operation.....84
   a) Step 1 – DNS Query Interception 90
   b) Step 2 – Connection Interception and Setup 98
   c) Step 3 – Encrypted Channel Communications.....106
9. MultiProxy 107
10. Secure Extranet Explorer 111
11. Domains Names of Hosts on the Private Networks Protected by an Aventail Extranet Server.....114

B. U.S. Patent No. 6,496,867 to Beser.....115

1. Overview of Beser.....115
2. Basic Components.....119
   d) Originating and Terminating End Devices.....120
   e) First and Second Network Devices 122
   f) Trusted-Third-Party Network Device 125
3. IP Tunnel.....128
4. Establishing an IP Tunnel 129
   g) Request Containing A Unique Identifier.....130
   h) Negotiation of Private IP Addresses 137

C. RFC 2401, the IPsec Protocol 141
1. Overview of RFC 2401 141
2. Implementing Aventail Using the RFC 2401 Protocol 151
3. Incorporating RFC 2401 into the Beser System 158
D. U.S. Patent No. 5,237,566 to Brand (Ex. 1012) 165
1. Overview of Brand 165
2. Combination of Aventail and Brand 166
3. Combination of Beser and Brand 168
E. RFC 2543, the Session Initiation Protocol 169
1. Overview of RFC 2543 169
2. Combining Aventail with RFC 2543 170
F. Additional Prior Art Combinations 171
1. Incorporating Beser with RFC 2401 with Brand 171
2. Combining Aventail, RFC 2401, and RFC 2543 172
3. Combining Aventail, RFC 2401, and Brand 173

I. INTRODUCTION

A. Engagement

1. I have been retained by counsel for Apple Inc. as an expert witness in the above-captioned proceeding. I have been asked to provide an opinion regarding the state of the art of the technology described in U.S. Patent Nos. 8,868,705 ("the '705 Patent") (Exhibit 1001) and 8,850,009 ("the '009 Patent") (Exhibit 1003). I have been asked to provide a description of various references that I understand are prior art to these patents and to provide a discussion of the meaning of certain words and phrases in the claims of these patents.

B. Background and Qualifications

2. I am the Plastech Professor of Computer Science at Brown University. My research interests include computer security, applied cryptography, analysis, design, and implementation of algorithms, graph drawing and computational geometry. I have published six textbooks and more than 240 peer-reviewed research articles in the above areas. I have given more than 70 invited lectures worldwide. I am a fellow of ACM, AAAS, and IEEE. I have received a Technical Achievement Award from the IEEE Computer. I am listed among the 360 most cited computer science authors worldwide by Thomson Scientific, Institute for Scientific Information (ISI). My research has been funded by ARO, DARPA, NATO, NSF, and several industrial sponsors (including Google, Microsoft,

NetApp, and Sun Microsystems). I received my Ph.D. degree in electrical and computer engineering from the University of Illinois at Urbana-Champaign in 1988.

3. I have extensive research and educational experience in computer and network security. I have developed and taught graduate and undergraduate computer security courses at Brown. I have coauthored a widely adopted textbook on computer security that includes two chapters (over one hundred pages) on network security, covering topics such as network layers, TCP/IP, DNS, firewalls, tunneling, and IPsec, wireless networks, and virtual private networks.

4. My research spans a broad range of topics in security and privacy, including cryptographic foundations, access control, authentication, data security and privacy, network security, email and web security, database security, application security, cloud computing security, and the visualization of security. Also, I have been an early developer of distributed systems that provide client-server applications over the internet.

5. I am often invited by the National Science Foundation to evaluate grant proposals on computer and network security. I am also regularly asked by my university and by peer institutions to give an opinion on the research in computer and network security by faculty candidates considered for hiring, tenure and

promotion. Finally, I am routinely serving in the program committees of highly selective international conferences in computer and network security.

6. My CV is included as an appendix to this report.

C. Compensation and Prior Testimony

7. I am being compensated at a rate of $500 per hour for my work in this matter. I am being reimbursed for reasonable and customary expenses associated with my work in this investigation. My compensation is not contingent on the outcome of this matter or the specifics of my testimony.

8. Within the last five years, I have testified by deposition in the matter of Dustan et al. v. comScore, Inc., a software privacy case where I served as an expert witness for comScore, Inc.,

D. Information Considered

9. My opinions are based on my years of education, research, and experience, as well as my investigation and study of relevant materials. In forming my opinions, I have considered the materials I identify in this report and those listed in Appendix A.

10. I may rely upon these materials and/or additional materials to respond to arguments raised by the Patent Owner. I may also consider additional documents and information in forming any necessary opinions — including documents that may not yet have been provided to me.

11. My analysis of the materials produced in this investigation is ongoing, and I will continue to review any new material as it is provided. This report represents only those opinions I have formed to date. I reserve the right to revise, supplement, and/or amend my opinions stated herein based on new information and on my continuing analysis of the materials already provided.

II. LEGAL STANDARDS FOR PATENTABILITY

12. Certain basic legal principles have been explained to me by counsel for Apple. Below, I have recorded these legal standards as they were explained to me.

13. First, I understand that for an invention claimed in a patent to be found patentable, it must be, among other things, new and not obvious from what was known before the invention was made.

14. I understand the information that is used to evaluate whether an invention is new and not obvious is generally referred to as "prior art" and generally includes patents and printed publications (e.g., books, journal publications, articles on websites, product manuals, etc.).

15. I understand that in this proceeding Apple has the burden of proving that the claims of the '705 and '009 Patents are anticipated by or obvious from the prior art by a preponderance of the evidence. I understand that "a preponderance

of the evidence" is evidence sufficient to show that a fact is more likely true than it is not.

16. As I discuss further in the claim construction section below, I understand that the claims of the '705 and '009 Patents must be given their broadest reasonable construction consistent with the patent specification. However, I also understand that it is possible that the '705 Patent will expire before the Board in these proceedings issues a final decision, in which case I understand that it is possible the Board may apply the *Phillips* standard of claim construction in that final decision. For this reason, I provide my understanding of that claim construction standard below. I understand that the claims, after being construed, are then to be compared to the information in the prior art.

17. I understand that in this proceeding, the information that may be evaluated is limited to patents and printed publications.

18. I understand that there are two ways in which prior art may render a patent claim unpatentable. First, the prior art can be shown to "anticipate" the claim. Second, the prior art can be shown to have made the claim "obvious" to a person of ordinary skill in the art. My understanding of the two legal standards is set forth below.

A. Anticipation

19. I understand that the following standards govern the determination of whether a patent claim is "anticipated" by the prior art.

20. I understand that the "prior art" includes patents and printed publications that existed before the earliest filing date (the "effective filing date") of the claim in the patent. I also understand that a patent will be prior art if it was filed before the effective filing date of the claimed invention, while a printed publication will be prior art if it was publicly available before that date.

21. I understand that, for a patent claim to be "anticipated" by the prior art, each and every requirement of the claim must be found, expressly or inherently, in a single prior art reference as recited in the claim. I understand that claim limitations that are not expressly described in a prior art reference may still be there if they are "inherent" to the thing or process being described in the prior art. For example, an indication in a prior art reference that a particular process complies with a published standard would indicate that the process must inherently perform certain steps or use certain data structures that are necessary to comply with the published standard.

22. I understand that if a reference incorporates other documents by reference, the incorporating reference and the incorporated reference(s) should be treated as a single prior art reference for purposes of analyzing anticipation.

23. I understand that it is acceptable to consider evidence other than the information in a particular prior art document to determine if a feature is necessarily present in or inherently described by that reference.

B. Obviousness

24. I understand that a claimed invention is not patentable if it would have been obvious to a person of ordinary skill in the field of the invention at the time the invention was made.

25. I understand that the obviousness standard is defined in the patent statute (35 U.S.C. § 103(a)) as follows:

26. A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

27. I understand that the following standards govern the determination of whether a claim in a patent is obvious.

28. I understand that to find a claim in a patent obvious, one must make certain findings regarding the claimed invention and the prior art. Specifically, I

understand that the obviousness question requires consideration of four factors (although not necessarily in the following order):

- The scope and content of the prior art;
- The differences between the prior art and the claims at issue;
- The knowledge of a person of ordinary skill in the pertinent art; and
- Whatever objective factors indicating obviousness or non-obviousness may be present in any particular case.

29. In addition, I understand that the obviousness inquiry should not be done in hindsight, but must be done using the perspective of a person of ordinary skill in the relevant art as of the effective filing date of the patent claim.

30. I understand the objective factors indicating obviousness or non-obviousness may include: commercial success of products covered by the patent claims; a long-felt need for the invention; failed attempts by others to make the invention; copying of the invention by others in the field; unexpected results achieved by the invention; praise of the invention by those in the field; the taking of licenses under the patent by others; expressions of surprise by experts and those skilled in the art at the making of the invention; and the patentee proceeded contrary to the accepted wisdom of the prior art. I also understand that any of this evidence must be specifically connected to the invention rather than being associated with the prior art or with marketing or other efforts to promote an invention. I am not presently aware of any evidence of "objective factors"

suggesting the claimed methods are not obvious, and reserve my right to address any such evidence if it is identified in the future.

31. I understand the combination of familiar elements according to known methods is likely to be obvious when it does no more than yield predictable results. I also understand that an example of a solution in one field of endeavor may make that solution obvious in another related field. I also understand that market demands or design considerations may prompt variations of a prior art system or process, either in the same field or a different one, and that these variations will ordinarily be considered obvious variations of what has been described in the prior art.

32. I also understand that if a person of ordinary skill can implement a predictable variation, that variation would have been considered obvious. I understand that for similar reasons, if a technique has been used to improve one device, and a person of ordinary skill in the art would recognize that it would improve similar devices in the same way, using that technique to improve the other device would have been obvious unless its actual application yielded unexpected results or challenges in implementation.

33. I understand that the obviousness analysis need not seek out precise teachings directed to the specific subject matter of the challenged claim, but instead can take account of the "ordinary innovation" and experimentation that

does no more than yield predictable results, which are inferences and creative steps that a person of ordinary skill in the art would employ.

34. I understand that sometimes it will be necessary to look to interrelated teachings of multiple patents; the effects of demands known to the design community or present in the marketplace; and the background knowledge possessed by a person having ordinary skill in the art. I understand that all these issues may be considered to determine whether there was an apparent reason to combine the known elements in the fashion claimed by the patent at issue.

35. I understand that the obviousness analysis cannot be confined by a formalistic conception of the words "teaching, suggestion, and motivation." I understand that in 2007, the Supreme Court issued its decision in *KSR Int'l Co. v. Teleflex, Inc.*, 550 U.S. 398 (2007), where the Court rejected the previous requirement of a "teaching, suggestion, or motivation to combine" known elements of prior art for purposes of an obviousness analysis as a precondition for finding obviousness. It is my understanding that *KSR* confirms that any motivation that would have been known to a person of skill in the art, including common sense, or derived from the nature of the problem to be solved, is sufficient to explain why references would have been combined.

36. I understand that a person of ordinary skill attempting to solve a problem will not be led only to those elements of prior art designed to solve the

same problem. I understand that under the *KSR* standard, steps suggested by common sense are important and should be considered. Common sense teaches that familiar items may have obvious uses beyond the particular application being described in a reference, that if something can be done once it is obvious to do it multiple times, and in many cases a person of ordinary skill will be able to fit the teachings of multiple patents together like pieces of a puzzle. As such, the prior art considered can be directed to any need or problem known in the field of endeavor as of the effective filing date and can provide a reason for combining the elements of the prior art in the manner claimed. In other words, the prior art does not need to be directed towards solving the same problem that is addressed in the patent. Further, the individual prior art references themselves need not all be directed towards solving the same problem.

37. I understand that an invention that might be considered an obvious variation or modification of the prior art may be considered non-obvious if one or more prior art references discourages or leads away from the line of inquiry disclosed in the reference(s). A reference does not "teach away" from an invention simply because the reference suggests that another embodiment of the invention is better or preferred. My understanding of the doctrine of teaching away requires a clear indication that the combination should not be attempted (e.g., because it would not work or explicit statements saying the combination should not be made).

38. I understand that a person of ordinary skill is also a person of ordinary creativity.

39. I further understand that in many fields, it may be that there is little discussion of obvious techniques or combinations, and it often may be the case that market demand, rather than scientific literature or knowledge, will drive design trends. When there is such a design need or market pressure to solve a problem and there are a finite number of identified, predictable solutions, a person of ordinary skill has good reason to pursue the known options within their technical grasp. If this leads to the anticipated success, it is likely the product not of innovation but of ordinary skill and common sense. In that instance the fact that a combination was obvious to try might show that it was obvious. The fact that a particular combination of prior art elements was "obvious to try" may indicate that the combination was obvious even if no one attempted the combination. If the combination was obvious to try (regardless of whether it was actually tried) or leads to anticipated success, then it is likely the result of ordinary skill and common sense rather than innovation.

III. THE '705 AND '009 PATENTS

A. Effective Filing Dates

1. Effective Filing Date of the '705 Patent

40. I have been informed that the '705 Patent issued from U.S. Application No. 13/615,557, which was filed on September 13, 2012. I have further been informed that the '557 application is a continuation of Application No. 13/049,552 (issued as U.S. Patent No. 8,572,247), which is a continuation of Application No. 11/840,560 (issued as U.S. Patent No. 7,921,211), which is a continuation of Application No. 10/714,849 (issued as U.S. Patent No. 7,418,504), which is a continuation of Application No. 09/558,210, filed April 26, 2000, and now abandoned, which is a continuation-in-part of Application No. of 09/504,783, filed on February 15, 2000 (issued as U.S. Patent No. 6,502,135), which is a continuation-in-part of Application No. 09/429,643, filed on October 29, 1999 (issued as U.S. Patent No. 7,010,604). The '210, '783 and '643 applications also claim priority to 60/106,261, filed October 30, 1998 and 60/137,704, filed June 7, 1998.

41. Based on my review of these applications, I believe the two independent claims of the '705 patent (claims 1 and 21) rely on information that was first included in the '783 application. I therefore understand that the priority date for these claims is the date that application was filed, namely: **February 15, 2000**. Because all of the other claims in the patent are dependent claims that rely

on claims 1 and 21, and because I have been informed that a patent claim cannot claim an earlier priority date than a claim from which that claim "depends," it is my understanding that the priority date for all of the claims of the '705 patent is February 15, 2000.

42. For example, I note that claim 1 of the '705 patent requires "intercepting … a request to look up an Internet Protocol (IP) address corresponding to a ***domain name***," while claim 21 specifies "[a] system … including … a server configuration arranged to: intercept … a request to look up an Internet Protocol (IP) address corresponding to a ***domain name*** … ." Based on my review of the documents, no application filed prior to the '783 application mentions the term "domain name" or otherwise provides a description of the techniques that appear in the '705 patent claims.

43. I also have been informed (and reviewed documents that show that) in proceedings involving related patents (the '135, '504, '151, '211, '274 and '697 patents), the Patent Owner has not disputed that claims including the words "domain name" have an effective filing date of at least February 15, 2000. *See, e.g.*, Patent Owner Preliminary Oppositions in IPR2013-00348, -00349, -00354, -00375 to -00378, -00393, -00394, -00397, and -00398, as well as IPR2014-00237, -00238, -00403, -00404, and -00610; *see also Inter Partes* Reexamination Nos. 95/001,682, 95/001,679, 95/001,697, 95/001,714, 95/001,788, and 95/001,789.

44. Thus, the effective filing date of the ’705 patent claims is not earlier than **February 15, 2000**.

2. Effective Filing Date of the ’009 Patent

45. I have been informed that the ’009 patent issued from Application No. 13/911,792 filed on June 6, 2013. I have further been informed that the ’792 application is a continuation of Application No. 13/903,788, filed May 28, 2013, which is a continuation of Application No. 13/336,790 (issued as U.S. Patent No. 8,458,341), which is a continuation of Application No. 13/049,552 (issued as U.S. Patent No. 8,572,247), which is a continuation of Application No. 11/840,560 (issued as U.S. Patent No. 7,921,211), which is a continuation of Application No. 10/714,849 (issued as Patent No. 7,418,504), which is a continuation of Application No. 09/558,210, filed April 26, 2000, and now abandoned, which is a continuation-in-part of Application No. of 09/504,783, filed on February 15, 2000 (issued as U.S. Patent No. 6,502,135), which is a continuation-in-part of Application No. 09/429,643, filed on October 29, 1999 (issued as U.S. Patent No. 7,010,604). The ’210, ’783 and ’643 applications also claim priority to 60/106,261, filed October 30, 1998 and 60/137,704, filed June 7, 1998.

46. Based on my review of these applications, I believe that the two independent claims of the ‘009 patent (claims 1 and 14) rely on information that was first included in the ’783 application. I therefore understand that the priority

date for these claims is the date that application was filed: **February 15, 2000**. Because all of the other claims in the patent are dependent claims that rely on claims 1 and 14, and because I have been informed that a patent claim cannot claim an earlier priority date than a claim from which that claim "depends," it is my understanding that the priority date for all claims of the '009 patent is February 15, 2000.

47. For example, I note that claim 1 of the '009 patent requires that a network device "send a ***domain name service (DNS) request***…" and "receive, following interception of the ***DNS request***," while claim 14 requires "sending a ***domain name service (DNS) request***…" and "receiving, following interception of the ***DNS request*** . . . ." Based on my review of the documents, no application filed prior to the '783 application uses the terms "domain name," "domain name service" and "DNS request," or otherwise provides a description of the techniques that appear in the '009 patent claims.

48. Furthermore, I have also been informed (and reviewed documents that show) that in proceedings involving the related '135, '504, '151, '211, '274 and '697 patents, there has been no dispute that claims including the words "domain name" or "domain name service" have an effective filing date of at least February 15, 2000. *See, e.g.*, Patent Owner Preliminary Oppositions in IPR2013-00348, -00349, -00354, -00375 to -00378, -00393, -00394, -00397, and -00398, as well as

49. Accordingly, it is my understanding that the effective filing date of the ’009 patent claims is no earlier than **February 15, 2000**.

B. Overview of The ‘705 and ‘009 Patents

50. The ’705 and ‘009 patents are generally focused on something called “Tunneled Agile Routing Protocol” (TARP), which these patents describe as a mechanism for secure network communication. Ex. 1001 (705 patent) at 3:16-19; Ex. 1003 (009 patent) at 3:20-23. As it is described in the patents, TARP provides secure and anonymous communications through several different mechanism: 1) tunneling; 2) an IP address hopping scheme where the IP addresses for computers and routers using the this scheme may change over time; 3) and other security techniques (such as using decoy data, fixed packet sizes, and session key encryption). Ex. 1001 (705 patent) at 1:35-38, 3:16-6:9; Ex. 1003 (009 patent) at 1:38-40, 3:20-6:13.

51. There are two short sections of the ’705 and ‘009 patents (roughly corresponding to columns 39-42 and 49-53 in each patent) that were added in a continuation-in-part application for each patent that was filed in February 2000. I will refer to these sections as the "DNS sections" of the patents. The DNS sections

refer to a technique for setting up secure communications in response to a DNS request specifying a secure destination. *See* Ex. 1001 (705 patent) at 38:62-41:53, 48:63-55:36; Ex. 1003 (009 patent) at 39:36-42:29, 49:41-53:49.

52. I have been informed that, in proceedings involving related patents, Patent Owner has asserted that the DNS sections provide written description support for claim terms involving domain names, DNS requests, requests to look up network addresses, and DNS servers. These are concepts that are addressed by the claims of the '705 patent.

53. The DNS sections describe modifying a "conventional DNS server" to allow that server to create virtual private networks. *See* Ex. 1001 (705 patent) at 39:58-40:22; Ex. 1003 (009 patent) at 40:29-57. According to these sections, the "modified DNS server" (Ex. 1001 (705 patent) at 39:62-64, Ex. 1003 (009 patent) at 40:33-34) will intercept a DNS request to look up a network address associated with a domain name. *See* Ex. 1001 (705 patent) at 38:65-67; 39:41-5; 40:1-3; Ex. 1003 (009 patent) at 39:33-38, 40:11-28; 40:39-41. The modified DNS server will then determine whether a secure site has been requested by, for example, checking an "internal table of such sites." *See* Ex. 1001 (705 patent) at 40:1-19; Ex. 1003 (009 patent) at 40:39-57. One of ordinary skill in the art would have understood that when the specification references an "internal table," it is referencing any one of a number ways to represent structured data, such as an actual table, a database, a

flat file, a map, etc. If it is determined that the DNS request corresponds to a secure site (e.g., because the name in the DNS request matches an entry in the table or database), the modified DNS server will then perform additional steps to establish a "virtual private network" with the specified secure site. *See* Ex. 1001 (705 patent) at 40:59-41:9, 51:29-35; Ex. 1003 (009 patent) at 41:31-49, 52:7-13.

54. The DNS sections indicate that the process of establishing a virtual private network can include conventional devices such as personal computers running web browsers, proxy servers, intermediate routers, and web servers. Ex. 1001(705 patent) at 39:58-67, 49:10-20, 52:20-26; Ex. 1003 (009 Patent) at 40:29-38, 49:55-65, 52:65-53:4.

55. The '705 and '009 patents also describe several optional features of the secure communication technique described in the DNS sections, such as using "IP hopblocks" to create a VPN, or incorporating user authentication. Ex. 1001 (705 patent) at 39:47-51, 39:56-57, 41:2-9, 51:43-56; Ex. 1003 (009 patent) at 40:18-22, 40:27-28, 41:42-49, 52:21-34. The patents also describe several optional configurations of the "modified DNS system," such as a standalone DNS server and a system that incorporates a DNS server, a DNS proxy server and a gatekeeper. Ex. 1001 (705 patent) at 40:30-42; Ex. 1003 (009 patent) at 41:1-14.

C. The Prosecution History of The '705 and '009 Patents

1. The '705 Patent

56. I understand that during prosecution of the '557 application, all claims were initially rejected under 35 U.S.C. § 103 as obvious based on U.S. Patent No. 5,898,830 ("Wesinger" – Ex. 1044), as well as on obviousness-type double patenting grounds. *See* Ex. 1002 (705 Patent File History) at 261-268. I understand that Patent Owner overcame these rejections (at least in part) by amending the claims to add (among other things) a step of "intercepting . . . a request to look up an Internet Protocol (IP) address corresponding to a domain name . . ." Ex. 1002 (705 Patent File History) at 338. I also understand that Patent Owner argued that Wesinger could be distinguished from the claims because the Wesinger process acts upon a "connection request" instead of a "request to look up [an] IP address." *Id.* at 343-348. I understand Patent Owner also filed terminal disclaimers to overcome double patenting rejections. Ex. 1002(705 Patent File History) at 408, 414-415. I further understand the Patent Examiner stated in a reasons for allowance that Wesinger and Kiuchi[1] "may not clearly disclose" the method steps recited in the independent claims. *See* Ex. 1002 (705 Patent File History) at 585-586 (quoting entirety of claim language).

---

[1] Takahiro Kiuchi and Shigekoto Kaihara, *C-HTTP - The Development of a Secure, Closed HTTP-based Network on the Internet* (February 1996).

2. The '009 Patent

57. I understand that during prosecution of the '792 application, all claims were initially rejected on obviousness-type double patenting grounds. *See* Ex. 1004 (009 Patent File History) at 230-237. In the same Office Action, the Examiner made the following observations regarding the claims' allowability over Kiuchi[2] and U.S. Patent No. 5,898,830 ("Wesinger"):

> **In Examiner's opinion, both Kiuchi and Wesinger may not clearly disclose the** feature of *"intercepting a domain name service (DNS) request to look up a network address of a second network device based on an identifier associated with the second network device* and determining whether or not to establish a secure communication connection over the encrypted communication link". Moreover, both Kiuchi and Wesinger may not clearly disclose "send a domain name service (DNS) request to look up a network address of a second network device based on an identifier associated with the second network device; receive, following interception of the DNS request and a determination that the second network device is available for the secure communications service: (1) an indication that the second network device is available for the secure communications service, (2) the requested network address of the second network device, and (3) provisioning information for an encrypted communication link".
>
> Moreover, in Examiner's opinion, Examiner believes that the requested is intercepted and determined before the request reached the firewall/DNS server.

Ex. 1004 (009 Patent File History) at 237-238.

---

[2] Ex. 1041: Takahiro Kiuchi and Shigekoto Kaihara, *C-HTTP - The Development of a Secure, Closed HTTP-based Network on the Internet* (February 1996).

58. I understand that Patent Owner overcame double patenting rejections by filing terminal disclaimers with respect to copending Application Nos. 13/903,788 and 13/618,966, as well as Patent No. 7,933,990 and the 7,188,180 and 8,051,181 patents. Ex. 1004 (009 Patent File History) at 370,373, 377, 385, 389. No Reasons for Allowance were included in the Notice of Allowance.

D. Construction of Terms Used in the '705 and '009 Patent Claims

1. Background on the Broadest Reasonable Interpretation

59. I understand that in an *inter partes* review proceeding of an unexpired patent the claims of the patent are to be given their broadest reasonable interpretation in light of the specification.

60. I also understand that, under the broadest reasonable interpretation standard, the claim terms must be evaluated using the ordinary meaning of the words being used in those claims from the perspective of a person of ordinary skill in the art in light of the specification.

61. I also understand that in an *inter partes* review proceeding of an expired patent the claims of the patent may be interpreted under what I understand are the traditional *Phillips* standards – with *Phillips* referring to the case *Phillips v. AWH Corp*, 415 F. 3d 1303 (Fed. Cir. 2005) – applied in district court litigation, rather than given their broadest reasonable interpretation in light of the specification.

62. I also understand that, under the *Phillips* standards, one should look first at the intrinsic record of the patent, including the claims, the specification and the prosecution history, and that terms are normally given their plain and ordinary meanings to one of skill in the art when read in the context of that intrinsic evidence.

63. I also understand that, under both the broadest reasonable interpretation standard and *Phillips* standards, where a patent applicant provides an explicit definition of a claim term in the specification that definition may control the interpretation of that term in the claim.

64. I also understand that, under the *Phillips* standards, limitations from the embodiments in the specification are generally not read into the claims, and, for a patent applicant's statement during prosecution to amount to a disclaimer of claim scope, any such statement must be a clear and unmistakable disavowal of that scope.

65. I understand that the ’705 and ’009 Patents are not expired, so their claims must be given the broadest reasonable interpretation consistent with the specification.

2. Broadest Reasonable Interpretation of Terms of the ’705 Patent

a) “intercepting . . . a request”

66. There are two independent claims (claim 1 and 21) in the ’705 patent and each uses the phrase “intercept[ing] . . . a request.” While claim 1 uses the phrase “intercepting … a request” and claim 21 uses the term “intercept … a request,” I do not believe that there is any substantive difference between these two phrases apart from the form of the verb.

67. I have been informed that, in a proceeding involving U.S. Patent No. 8,504,697 (the ’697 patent), the Board interpreted the term “intercepting” as including the process of “receiving a request pertaining to a first entity at another entity.” IPR2014-00237, Paper 15 at 13 (May 14, 2014). I note that the '697 patent has the same DNS sections as the '705 and '009 patents. I have also been informed the Board provided the additional explanation that “intercepting” a request involves “receiving and acting on” a request, the request being “intended for” receipt at a destination other than the destination at which the request is intercepted. *Id.* at 12. In my opinion, this construction is consistent with the ’705 patent disclosure.

68. There is no definition of the phrase “intercepting . . . a request” in the ’705. I believe based on how the term "intercepting" is being used in the patents, one of ordinary skill in the art reading the patents would understand the term

"intercepting" to mean receiving a request at a device other than the device for which the request was intended. Based on my review of the specification, the most germane discussion in the patent of this concept relates to a DNS proxy that "intercepts" all DNS lookup functions in order to determine whether access to a secure site has been requested. Ex. 1001 (705 patent) at 40:1-7, Figs. 26 & 27. The specification explains that while the DNS server (2609) ordinarily would receive and resolve domain name requests, DNS requests are instead routed to the DNS proxy. Ex. 1001 (705 patent) at 40:1-3. The patents indicate the DNS proxy and DNS server can, in one configuration, be deployed on separate computers. Ex. 1001 (705 patent) at 40:38-42. It is therefore my opinion that the '705 patent uses the term "intercept" to mean receipt of a message by a DNS proxy server instead of the intended destination (the DNS server).

69. Accordingly, the broadest reasonable interpretation of the term "intercepting . . . a request" includes a proxy computer or device "***receiving a request pertaining to a first entity at another entity***."

b) "domain name"

70. Both independent claims of the '705 patent use the term "***domain name***." There is no explicit definition for the term "domain name" in the '705 patent. In my opinion, one of ordinary skill in the art would understand the term "domain name" to have its ordinary meaning of being a hierarchical sequence of

one or more names arranged in decreasing order of specificity that corresponds to a resource on a network with an IP address. I have been informed, however, that VirnetX has argued in other proceedings that "domain name" means "a name corresponding to an IP address." *See, e.g.*, Ex. 1042 (VirnetX Claim Construction Brief) at 14-15. This alternate construction is more general but I believe it also represents what one of ordinary skill in the art would understand a domain name to be, because domain names are names that are used by the Domain Name Service to lookup IP addresses. Therefore, it is my opinion that the broadest reasonable construction of "domain name" is "***a name corresponding to an IP address***."

c) "secure domain name"

71. Claims 3 and 10 use the phrase "***secure domain name***," and state that a "secure domain name" is a type of "domain name." The concept of a secure domain name has been in introduced in several patents issued to Larson and others (including the '705 and '009 patents) but it is not a standard technical term that would be understood by someone skilled in the art as having a single, accepted meaning. I note the term "secure" is typically used in reference to devices, services, or networks, but not to network addresses themselves. Thus, for example, a "secure computer network address" would include a network address for a secure computer, a network address for a secure service, or an address in a secure computer network.

72. I have been informed that, in a related proceeding involving the '180 patent, the Board determined that "a 'secure domain name' is a name that corresponds to a secure computer network address." IPR2015-00481, Paper 11 at 8 (Sept. 3, 2014).

73. In the '705 patent and claims, a "secure domain name" is described as a domain name corresponding to the network address of a secure server (3320). *See* Ex. 1001 (705 patent) at 51:6-42. The specification further notes that "[e]ach secure computer network address is based on a non-standard top-level domain name, such as .scom, .sorg, .snet, .sedu, .smil and .sint." Ex. 1001 (705 patent) at 7:39-42. In this example, the specification seems to equate a secure computer network address with a non-standard top-level domain name. However, as the term is used in the '705 patent and the claims of this patent, it has a more general meaning of being a name that corresponds to a particular device on a secure computer network (i.e., one that would have an address on that secure computer network). Therefore, I believe the broadest reasonable interpretation of the term "secure domain name" is "***a name that corresponds to a secure computer network address***."

d) "provisioning information"

74. Both of the independent claims in the '705 patent use the term "***provisioning information***," but no explicit definition of that term is provided in

the patent disclosure. Based on my review of the specification, the only passage that I can find that discusses the term "provision" in any way states: "VPN gatekeeper 3314 ***provisions*** computer 3301 and secure web server computer 3320, or a secure edge router for server computer 3320, thereby creating the VPN." Ex. 1001 (705 patent) at 51:32-35 (emphasis added); *see id.* at 51:57-60. The specifications of the patent explain that, as part of creating this VPN, the DNS proxy first determines that access to a secure site has been requested, and then transmits a message to a gatekeeper requesting creation of a VPN. Ex. 1001 (705 Patent) at 40:7-10, 40:66-41:2. The gatekeeper then is responsible for returning certain information (*i.e.*, an IP address and IP address "hopblocks") that the client computer and the target will use for secure communications in the VPN. *Id.* at 40:10-19.

75. I have been informed that in IPR2014-00481 – which involved the related '180 patent – there were claims at issue that included a step where provisioning information was to be provided for a "virtual private network" as opposed to the "encrypted communications channel" recited in the '705 patent. I have been informed that the Board construed the term "provisioning information" as "information that is provided to enable or to aid in establishing communications to occur in the VPN." PR2014-00481, Paper 11 at 11 (Sept. 3, 2014). In my opinion, this construction properly captures the substance of the disclosures that I

discuss directly above – namely, that provisioning involves providing information (an IP address and hopblocks) that is used to enable the creation of a VPN. The only difference for the claims of the '705 patent is that they reference an "encrypted communications channel" so a proper construction for "provisioning" must take into account encryption as well.

76. Accordingly, it is my opinion that the broadest reasonable interpretation of the phrase "provisioning information" in the '705 patent claims is "***information that enables communication in a virtual private network, where the virtual private network uses encryption***."

e) "modulated transmission link" / "unmodulated transmission link"

77. Claims 5, 11 and 27 of the '705 patent contain the phrase "***unmodulated transmission link,***" while claims 6, 12 and 28 use "***modulated transmission link***." Based on my review of the patent specification, neither of these term is given an express definition in the '705 patent.

78. I have been informed that in IPR2014-00237 – which involved the related '697 patent – the Board interpreted "modulation" to include "the process of encoding data for transmission over a medium by varying a carrier signal." Paper 15 at 14 (May 14, 2014).

79. In my opinion, the Board's construction is consistent with the '705 patent specification and the understanding of a person of ordinary skill in the art. To provide some context for interpreting these terms, I note that the '705 specification explains that transmission paths may be "logically separate paths contained within a broadband communication medium (*e.g.*, separate channels in an FDM, TDM, CDMA, or other type of modulated or unmodulated transmission link)." Ex. 1001 (705 patent) at 34:49-55.

80. To one of ordinary skill in the art, an "unmodulated" signal would generally be thought of as a baseband signal that is sent directly over the transmission medium without using that signal to modulate a carrier signal, whereas a "modulated" signal would involve modulating a carrier signal by using the baseband signal to allow for multiple communications over the communication link (i.e., frequency division multiplexing). In other words, one of ordinary skill in the art understand that the terms "unmodulated" and "modulated" refer to whether data is encoded for transmission over a physical medium by varying (that is, by "modulating") a carrier signal. Examples of modulated signals include data transmitted via a modem (*i.e.*, a "modulator-demodulator" device) and data transmitted via a cellular or cable network. Data transmitted via a baseband network, such as an Ethernet network, is an example of an unmodulated signal.

81. Therefore, it is my opinion that the broadest reasonable interpretation of "***modulated transmission link***" is "***a communication medium for transmitting data that is encoded by varying a carrier signal***." Also, it is my opinion that the broadest reasonable interpretation of "***unmodulated transmission link***" is "***a communication medium for transmitting data that is not encoded by varying a carrier signal***."

f) "phone"

82. The word "phone" is used in claims 8, 15, 30, and 32 of the '705 patent. No express definition is given for this term, but the specification mentions that "telephony" is an example of an "application program[]" that may be executed on a "computer node." Ex. 1001 (705 patent) at 21:21-28. The only other passage in the specification that could be read to relate to a "phone" is a reference to sending data over telephone *lines*. Ex. 1001 (705 patent) at 20:3-5, 34:49-55. Based on these passages in the specification, it is my opinion that one of ordinary skill would understand the term "phone" is not being limited to a physical handset connected to the public switched telephone network, but instead is broad enough to include any device (or hardware or software component) that can provide telephony functionality, which I believe is consistent with what one of ordinary skill in the art would have understood the ordinary meaning of "phone" to be.

83. Accordingly, the broadest reasonable interpretation of "phone" is therefore "***a device or component that can provide telephony functionality***."

3. Broadest Reasonable Interpretation of Terms of the '009 Patent

a) "domain name service (DNS) request"

84. Both independent claims (claims 1 and 14) in the '009 patent use the phrase "***domain name service (DNS) request***." Based on my review of the '009 patent, no definition of this term is provided. I have been informed that in IPR2014-00610 – which involved the related '151 patent – the Board construed "DNS request" to mean "a request for a resource corresponding to a domain name." Paper 9 at 6 (Oct. 15, 2014).

85. It is my opinion that the Board's construction is consistent with the meaning of this term based on the '009 patent specification, because the specification describes using DNS requests in order to determine a network address corresponding to a provided "web name" or "domain name." Ex. 1003 (009 patent) at 39:39-45, 40:39-45, 40:52-58. Accordingly, the broadest reasonable interpretation of the "domain name service (DNS) request" would be "***a request for a resource corresponding to a domain name***."

b) "interception of the DNS request"

86. Both independent claims of the '009 patent (claims 1 and 14) use the phrase "***interception of a DNS request***." As with the term "interception" for the '705 patent, no express definition is given in the specification for this term. For the same reasons given above with respect to the "intercepting" term for the '705 patent (see above at ¶¶66-69), it is my opinion that the broadest reasonable interpretation of this term includes a proxy computer or device "***receiving a DNS request pertaining to a first entity at another entity***."

c) "encrypted communication link"

87. Both of the independent claims (claims 1 and 14) in the '009 patent use the term "***encrypted communications link***," but the '009 patent does not provide an express definition for this term. I have been informed that the Board has not interpreted this exact term before in the related proceedings, but has had occasion to address the phrase "secure communication link" and "virtual private network communication link."

88. I have been told that in IPR2014-00237 – which involved the related '697 patent – the Board construed the phrase "secure communication link" to mean "a transmission path that restricts access to data, addresses, or other information on the path, generally using obfuscation methods to hide information on the path,

including, but not limited to, one or more of authentication, encryption, or address hopping." Paper 15 at 10 (May 4, 2014).

89. Likewise, I have been told that in IPR2014-00481 – which involved the related '180 patent – the Board construed "virtual private network communication link" to mean "a transmission path between two devices that restricts access to data, addresses, or other information on the path, generally using obfuscation methods to hide information on the path, including, but not limited to, one or more of authentication, encryption, or address hopping." Paper 11 at 6-7 (Sept. 3, 2014).

90. As with the '697 and '180 patent claims, the '009 patent claims specify that communication is performed over a "communication link," but the '009 claims further require this link to be "encrypted." It is my opinion that one of ordinary skill in the art would understand that this "communication link" resulting from using the techniques described in the specification (*see* Ex. 1003 (009 Patent) at 39:56-65) is a secure communication link that is secured via encryption.

91. Accordingly, it is my opinion that the broadest reasonable interpretation of "encrypted communication link" in claims of the '009 patent is "a transmission path that restricts access to data, addresses, or other information on the path at least by using encryption."

d) "provisioning information"

92. The two independent claims in the '009 patent recite the term "***provisioning information***." As with the '705 patent, no explicit definition of this term is provided in the '009 patent. For the same reasons I have described above with respect to the '705 patent (see above ¶¶74-76), it is my opinion that the broadest reasonable interpretation of the term "provisioning information" in the context of the '009 claims is "***information that enables communication in a virtual private network, where the virtual private network uses encryption***."

e) "secure communications service"

93. Both of the independent claims of the '009 patent (claims 1 and 14) use the term "***secure communications service***," but no express definition is provided for this term. I have been informed that in IPR2014-00237 – which involved the related '697 patent – the Board construed the term "secure communications service" as "the functional configuration of a network device that enables it to participate in a secure communication link with another network device." Paper 15 at 10 (May 14, 2014).

94. As I have mentioned already, "secure communication link" was interpreted by the Board to mean "a transmission path that restricts access to data, addresses, or other information on the path, generally using obfuscation methods to hide information on the path, including, but not limited to, one or more of

authentication, encryption, or address hopping." IPR2014-00237, Paper 15 at 10 (May 4, 2014).

95. In my opinion, these constructions are consistent with the '009 specification and the understanding of one of ordinary skill in the art. One of ordinary skill in the art reading the '009 patent would understand that the phrase "secure communications service" is being used to refer to the capacity of two computers to communicate securely. As an example, the '009 patent explains that a first network device "communicat[es] at least one of video data and audio data with the second network device using the secure communications service via the secure communication link." Ex. 1003 (009 patent) at 8:28-31, 8:45-48.

96. Therefore, it is my opinion that the broadest reasonable construction of the term "secure communications service" includes "the functional configuration of a network device that enables it to participate in a secure communications link with another computer or device."

f) "indication"

97. Both of the independent claims of the '009 patent (claims 1 and 14) require that a device receive "an indication" that a second device is "available" for a secure communications service. No express definition for this term is provided in the '009 specification.

98. I have been informed that in IPR2014-00614 – which involved the related '504 patent – the Board interpreted the term "indication" to mean "something that shows the probable presence or existence or nature of." Paper 9 at 12-13 (Oct. 15, 2014).

99. In my opinion, this construction is consistent with how that term is used in the '009 specification and how it would be understood by one of ordinary skill in the art. The specification states that when the DNS proxy determines that access to a secure site has been requested, the DNS proxy then forwards that request to a gatekeeper, and in response the client can receive a "resolved" IP address along with other provisioning information (such as IP "hopblocks") to be used to engage in secure communication with the secure site. Ex. 1003 (009 patent) at 40:39-57. Alternatively, the DNS proxy could return a "host unknown" error message, if the host cannot be found or if the user lacks the appropriate authorization credentials. Ex. 1003 (009 patent) at 40:62-65.

100. In my opinion, the specification's discussion of an icon that may be shown in a web browser indicating that a secure connection has been established (Ex. 1003 (009 patent) at 52:37-40) does not inform the meaning of "indication" because it is not related to a client receiving a message confirming that the secure target site is available for secure communications, but instead an indication that secure communications have been established.

101. Therefore, it is my opinion that the broadest reasonable interpretation of the term "indication" should include "***something that shows the probable presence or existence or nature of.***"

g) "virtual private network communication link"

102. Claims 8 and 21 state that the encrypted communication link "is part of a ***virtual private network communication link***." No express definition of the phrase "virtual private network communication link" is set out in the specification.

103. I have been informed that in IPR2014-00481 – which involved the related '180 patent – the Board construed "virtual private network communication link" to mean "a transmission path between two devices that restricts access to data, addresses, or other information on the path, generally using obfuscation methods to hide information on the path, including, but not limited to, one or more of authentication, encryption, or address hopping." Paper 11 at 6-7 (Sept. 3, 2014); *see also* IPR2014-00403, Paper 13 at 8-9 (Jul. 31, 2014) (wherein the term "virtual private network communication link" was construed in proceedings involving the related '274 patent). I have further been informed that the Board determined that the '180 patent employed various levels of security in a VPN that did not require encryption, such as authentication, or information or address hopping. *Id.* at 7.

104. In my opinion, the Board's construction of this term is consistent with the specification of the '009 patent and the understanding of one of ordinary skill

in the art. In the specification, a VPN communication link is made reference to in connection with a software module accessing a secure server. Ex. 1003 (009 Patent) at 52:35-36 ("software module 3309 accesses secure server 3320 through VPN communication link 3321."). This connection is illustrated in FIG. 33 of the '009 patent, where the communication link 3321 includes only the portion of the path between computer 3301 and server 3320 that is over network 3302. In other words, this VPN connection link need not be an end-to-end connection to the final target with which a remote device wants to communicate, but can be the connection between the remote device and the server that provides access to the private network.

105. Therefore, it is my opinion that the broadest reasonable interpretation of "virtual private network communication link" is "***a transmission path between two devices that restricts access to data, addresses, or other information on the path, generally using obfuscation methods to hide information on the path, including, but not limited to, one or more of authentication, encryption, or address hopping***."

h) "domain name"

106. Dependent claims 7 and 20 recite the term "***domain name***." As with the '705 patent, the '009 patent does not provide a definition for the term "domain name." For the reasons I have given above in my discussion of this term for the

'705 patent (see above ¶70), it is my opinion that the broadest reasonable construction of "domain name" is "***a name corresponding to an IP address***."

i) "modulation"

107. Claims 4, 5, 17 and 18 of the '009 patent use the term "***modulation***," but no definition for this term is provided in the patent. In IPR2014-00237 involving the '697 patent, the Board interpreted "modulation" to include "the process of encoding data for transmission over a medium by varying a carrier signal." Paper 15 at 14 (May 14, 2014).

108. For the reasons I have already provided above in my discussion of the terms "modulated transmission link" and "unmodulated transmission link" (see above ¶¶77-81) it is my opinion that the broadest reasonable interpretation of "modulation" is "***the process of encoding data for transmission over a medium by varying a carrier signal***."

E. Level of Ordinary Skill in the Art

109. I have been instructed that the claims of a patent are to be reviewed from the point of view of a hypothetical person of ordinary skill in the art at the time of the filing of the patent.

110. I believe a person of ordinary skill in the art in the field of the '705 and '009 patents would be someone who had a good working knowledge of networking protocols, including those employing security techniques, as well as

cryptographic methods and computer systems that support these protocols and techniques. The person would have gained this knowledge either through several years of practical working experience or through education and training, or by a combination of training and education. The person would also be very familiar with Internet standards related to communications and security, with software development techniques, and with a variety of client-server systems and technologies. In preparing this declaration, I have considered the issues from the perspective of a hypothetical person of ordinary skill in the art.

IV. TECHNICAL BACKGROUND

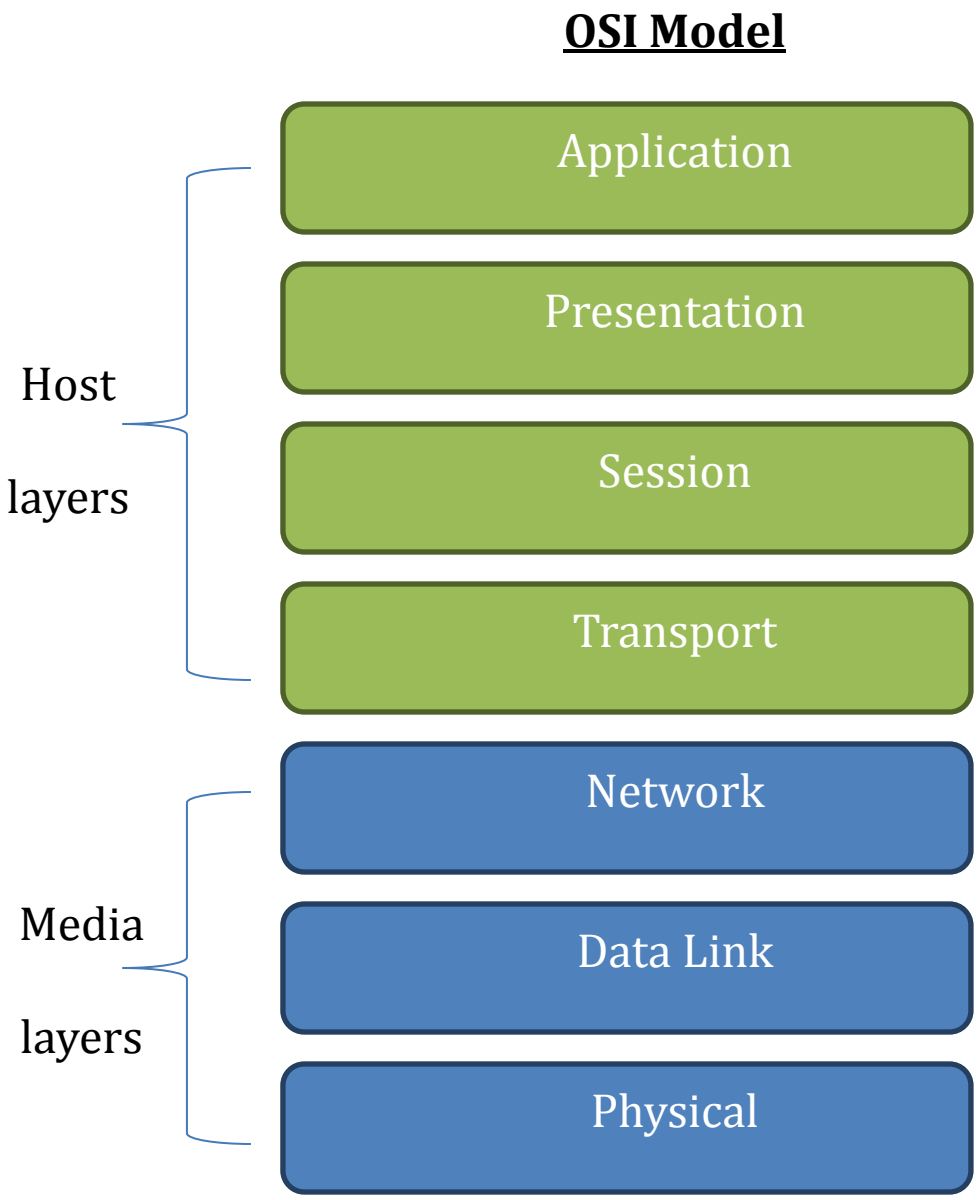
A. COMPUTER NETWORKS

j) The OSI Model

111. The complexity of the different types networks that could be connected together, the numerous standards that were being developed for different types of communication systems, and the rapid adoption and commercialization of networked systems provided a need for a foundational networking standard. The International Standards Organization (ISO), an organization of representatives of numerous national standards-setting bodies, develops, adopts, and promotes standards in many disparate areas, including networking. To allow the simultaneous development of standards and minimize the potential for conflict between these developing standards, the ISO adopted a convention for the many

functions that must be provided by a network. This convention is called the ISO Reference Model for Open Systems Interconnection (ISO/IEC 7498-1), or the "OSI model".

112. The OSI model describes seven different major functions of networks, network protocols, and network devices. These seven major functions are arranged in seven layers. The seven layers (from bottom to top) are the physical layer, the data link layer, the network layer, the transport layer, the session layer, the presentation layer, and the application layer. I have illustrated this below with a picture:



113. Each of these layers provides a well-defined set of services to the layer above it, through an abstraction called the Service Access Point (SAP). Standards developed for a particular layer that adhere to the OSI model can be connected, theoretically, to any standard that adheres to the OSI model for the layers above or below. The Internet is a spectacularly successful example of following this approach.

k) The Internet

114. The Internet is not a single, monolithic construction that one can point to and identify. Rather, the Internet is a combination of a vast number of independent public and private networks, sharing a common set of protocols and cooperating to connect billions of devices and users. In common usage, as well as in the understanding of one of ordinary skill in the art, almost any network connection to a public network is a connection to the Internet. Even those network connections wholly within a private network are connections to the Internet, as long as there is a path from the connected device through the private network to a public network.

115. Creating the path from a network connected device to the Internet is accomplished by processes operated by protocols at layers 1, 2, and 3 of the ISO Model. Layer 1, the Physical Layer, creates the electrical, optical, or electromagnetic signals that pass information between two devices. Layer 2, the Data Link Layer, utilizes one or more protocols to provide addressing to the devices connected by Layer 1 that will forward information in a local area network (LAN) by switching and bridging that information. Layer 3, the Network Layer, utilizes additional addressing and protocols to deliver information between multiple Layer 2 LANs through a process called routing.

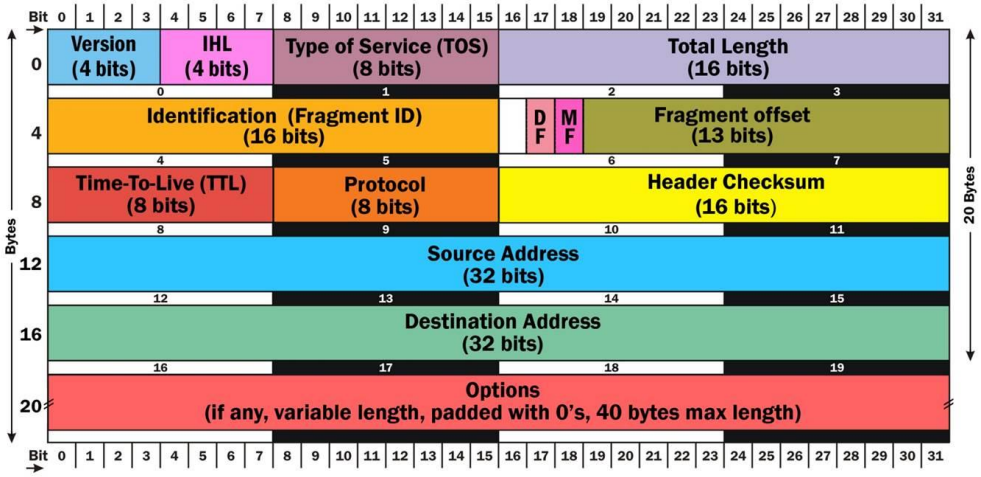
116. By connecting Layer 2 LANs with Layer 3 routing devices (routers), information from any device on any of the connected Layer 2 LANs can be communicated to any other device on those connected LANs. This is the process by which the Internet is constructed. To anyone of ordinary skill in the art, this would be common knowledge.

B. Internet Protocol Suite (TCP/IP)

117. The Internet protocol suite is a set of communications protocols that is used to send data over the Internet. The suite is generally referred to as TCP/IP, which stands for the Transmission Control Protocol and the Internet Protocol. Together these two protocols provide for end-to-end (computer-to-computer) communications by defining how data that is to be sent over the Internet is packaged, addressed, and routed. TCP/IP is a two "layer" protocol, with TCP being the higher layer and IP the lower layer. IP lies at the OSI network layer and is responsible for routing packets through the network, from a source node to a destination node, on a best effort basis, possibly using intermediate nodes. TCP lies at the OSI transport layer. It is responsible for establishing a reliable communication channel between processes on two nodes. Tasks performed by TCP include the ordering or packets and the retransmission of lost packets.

118. An IP packet comprises two pieces: a payload, which contains the data being sent; and a header, which contains the information necessary to route

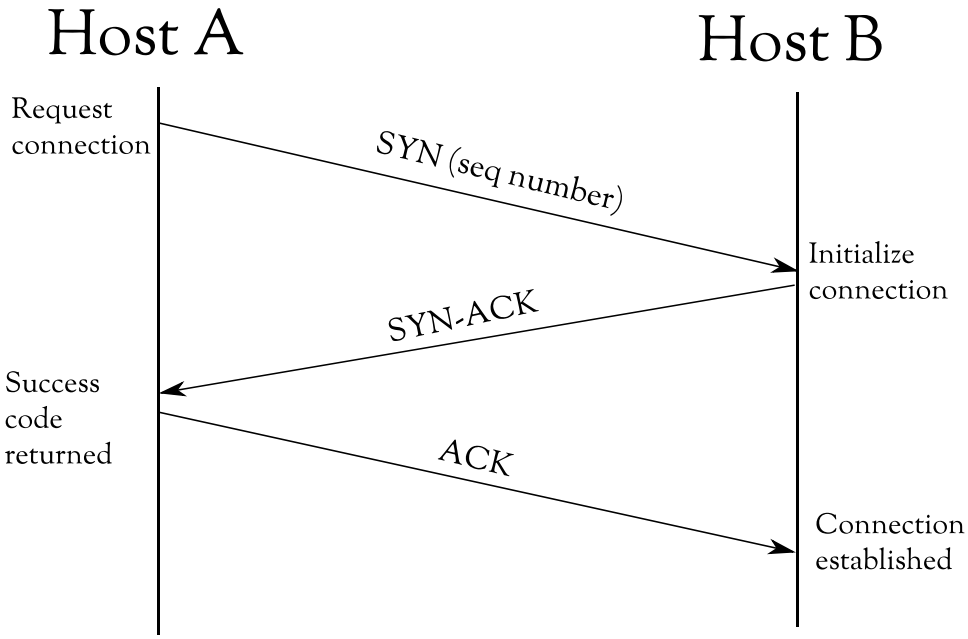
the message across the network. An illustration of a IP v4 packet header is given below:



119. As can be seen from this illustration, two of the fields in the header are the "source address" (the computer that is sending the message) and the "destination address" (the computer that the message is being sent to). These addresses – called Internet Protocol (IP) addresses – are 32 bit numbers that are used to identify a particular computer on the network. Though they are stored and process by a computer in binary, when an IP addressed in displayed in a human-readable format, it takes the form of a string of four dot-separated numbers between 0 and 255. An example of an IP address is 23.100.43.208.

120. For two devices to communicate using TCP, certain information must be exchanged in order for the TCP communication channel to be formed. The process by which this information is exchanged and a connection established is

called the TCP three-way "handshake." I have provided an illustration of this process below:



121. This handshake is called "three-way" because it involves the transmission of three messages used to negotiate the TCP channel parameters. For example, the SYN message includes port numbers identifying the communicating processes at the sender and receiver, an initial sequence number that will be incremented by 1 for each message sent, and with a maximum size for the next data packet to be received.

C. DOMAIN NAME SYSTEM (DNS)

122. As I explained above, data from one computer is sent to another computer over the Internet as a series of IP packets, where the source and

destination computers are identified by IP addresses stored in the IP packet. Of course, it is difficult for humans to remember IP addresses. Also, it is desirable to change from time to time the computer providing a certain service. Thus, humans and applications typically use names (hostnames) instead of IP addresses to specify the particular computer with which they wish to communicate. These hostnames need to be "resolved" into IP addresses so that messages can be appropriately addressed and routed. This technique of mapping an identifier to an address and sending a query using the identifier to obtain the associated address has been widely used since at least the 1970s in the computer networking field and would have been well known to those of ordinary skill in the art.

123. In the early days of ARPAnet, there were only a small number of computers networked together, and all name to address mappings were kept in a single text file, called HOSTS.TXT. This file had a very simple format and contained a table with each line listing the hostname, the IP address, and supported protocols.

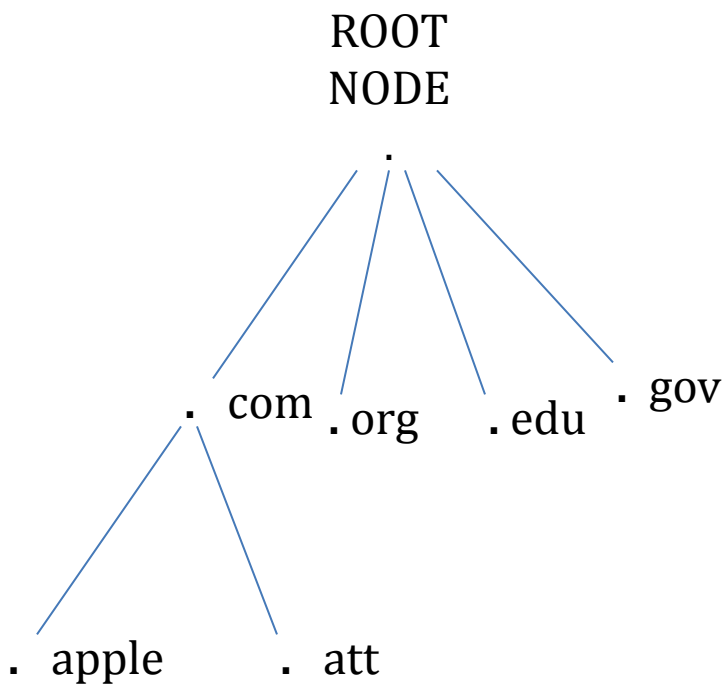
124. Each computer on the network had a copy of the HOSTS.TXT file and was responsible for locally resolving IP addresses for hostnames. The HOSTS.TXT file was maintained and distributed by SRI's Network Information Center (NIC). If changes needed to be to the HOSTS.TXT file (i.e., a new computer being added to the network) an email would be sent with the changes to

the NIC, those changes would need to be implemented, and then every one of the computers on the network would need to download the new HOSTS.TXT file.

125. Obviously, this process did not scale well, and as more and more computers become internetworked, the HOSTS.TXT model grew unworkable. A distributed solution was needed, and in 1984 two RFCs (882 and 883) were released that specified an architecture for a decentralized system called the Domain Name System. *See generally*, Ex. 1032 (RFC 882) and Ex. 1033 (RFC 883); In 1987 those RFCs were superseded by RFCs 1034 and 1035 (Exs. 1034 and 1035, respectively), which describes the Domain Name System that has been widely used ever since.

126. The Domain Name System (DNS) translates hostnames (e.g., www.apple.com) into IP address (e.g., 17.172.224.47) necessary to route communications to a destination. DNS is a distributed database that allows for address resolution via a client-server model. The servers in this model are programs called "name servers" that make information from the database available to client programs called "resolvers," which are responsible for creating queries and submitting them to a name server.

127. Much like a file system, the DNS database is structured like an inverted tree, with a "root" node at the top, with various branching subtrees (subdomains) beneath, as shown in the picture below:



128. Each node in the database is referred to as a "domain" and can have other nodes below it. A domain's "domain name" is the sequence of labels on the path from the domain itself up to the root node. In DNS, responsibility for each subdomain can be assigned separately.

D. NETWORK ENCRYPTION

129. Security and encryption have always been important considerations in the design and development of computer networks. Some of the earliest computer networks were designed to support military and defense operations and so securing and encrypting communications was paramount. And as computer networks grew and began to be used in the private sector, securing and encrypting communications remained important for purposes of privacy and protecting

confidential business information. Below I provide a brief summary of the two most common types of encryption that are used to protect network communications – symmetric encryption and asymmetric encryption (also called public/private key encryption).

1. Symmetric Encryption – DES & AES

130. Symmetric encryption, also known as private-key encryption, refers to a cryptographic technique whereby a secret "key" is used for both encryption and decryption. One very early example of this type of encryption was the Caesar cipher (or shift), where the "key" specified that each letter of the plaintext message was shifted a given number of positions down the alphabet (e.g., the key could specify that letters were to be shifted right 11 positions, meaning that the letter A would become L, B would become M, etc.).

131. Substantially more complex symmetric encryption algorithms began to be publicly used for encrypting communications over computer networks in the 1970s with the publication of the Draft Encryption Standard (DES). The symmetric encryption method currently recommend by the US government is the Advanced Encryption Standard (AES), which is based on the Rijndael cipher originally developed by Joan Daemen and Vincent Rijmen in 1998.

2. Asymmetric Key Encryption – Public/Private Keys

132. One of the fundamental problems with symmetric encryption schemes is that they require that both parties know (and keep secret) the encryption key, meaning that this key must be securely exchanged between the parties at some point. This leads to a chicken-and-egg problem whereby the distribution of keys to allow for secure communications itself required secure communications.

133. Whitfield Diffie and Martin Hellman proposed a solution to this problem of key distribution in their seminal 1976 paper "New Direction in Cryptography," wherein they proposed the use of asymmetric encryption.

134. Asymmetric encryption uses a pair of keys, called public key and private key, respectively, which are mathematically related. The private key is used to decrypt content that has been encrypted with the public key. In practice, each user (or computer) has its own key pair. The private key is kept secret (known only by the user) while the encryption key is publicly shared, meaning that there is no need for a secure channel to distribute keys – any user that wishes to securely communicate with another can simply obtain the other's public key over a public channel and then use that public key to encrypt communications sent to that other user. To reply, the other user can do the same, using the public key of the sender to encrypt a response.

135. An encryption system that uses public-private key pairs can be generally used to provide digital signature, which provide assurance of the author of a message and of the integrity of the message. Basically, a digital signature on a (plaintext) message is computed with the private key by applying the decryption algorithm to the message. Conversely, the signature can be verified using the public key and the encryption algorithm.

3. Key Exchange

136. There are several ways to distribute the key information required to use encryption techniques. Typically, most cryptographic systems include a trusted authority or trusted device that will certify or distribute key information. In a public key system, typically there are one or more trusted-third-party devices that serve as a certificate authority (CA). A CA issues certificates, which are digitally signed statements that associate a public key with an identity. See, e.g., RFC 2459 ("Internet X.509 Public Key Infrastructure Certificate and CRL Profile"). Using certificates, a party can be assured of the identity of other parties with which it exchanges encrypted messages.

137. Another technique is to use a Key Distribution Center or "KDC" to provide key information for transactions. One well-known example of a KDC is Kerberos. *See generally* Ex. 1045 (Kerberos). Each entity participating in the system will register a symmetric key with the KDC. When the KDC provides

session keys or authentication information to an entity, the KDC will encrypt the data with the entity's registered key. *Id.*

138. One common use of a KDC was to distribute symmetric session keys to be used to secure the communications between two devices in a session. To create a session key to be used by devices A and B, the KDC would create a random session key, encrypting a copy of it with A's registered key, encrypting a copy of it with B's registered key, and then providing the two encryptions of the session key to A and B. Alternatively, the KDC could communicate only with A, as follows. It would send to A both the session key encrypted with A's registered key and a message consisting of the session key and the identity of A encrypted with B's registered key. This message would then be forwarded by A to B. Ex. 1045 (Kerberos) at 3.

139. The Internet Key Exchange (IKE) protocol (RFC 2409) was well-known to those of ordinary skill in the art circa 2000. IKE describes how two devices can mutually agree on cryptographic methods and keys to be used in a subsequent secure communication session that is encrypted and/or authenticated. EX. 1046 (RFC 2409) at 2-3. IKE is a negotiation that consists of a series of messages exchanged between the two devices. At the end of the negotiation, the two parties have identified cryptographic methods (such as encryption systems, hash functions, and their parameters) and cryptographic keys for them. Ex. 1046

(RFC 2409) at 8-19. IKE ensures that only the two devices know the keys obtained at the end of the negotiation. Also, IKE provides mutual authentication of the two devices, *i.e.*, each device is assured of the identity of the other device, as mandated by the Internet Security Association and Key Management Protocol (ISAKMP). See Ex. 1047 (RFC 2408) at 8-9.

140. IKE supports several methods to achieve mutual authentication. These methods rely, in turn, on one of the following two approaches: (1) each device knows or can learn the public key of the other device; or (2) the two devices hold a previously shared secret key. EX. 1046 (RFC 2409) at 8-19. In the first approach, each device could obtain the public key of the other device from a trusted-third-party device. Alternatively, each device could send to the other party its public key plus a certificate attesting its validity signed by a trusted-third-party device, which serves a certificate authority (CA). In the second method, one or both devices could communicate with a trusted-third-party, which generates the shared key by operating as a key distribution center (KDC). Note that the session keys generated by the IKE negotiation are computed from random values picked by the two devices and cannot be derived from the keys used for authentication. In particular, neither an eavesdropper of the IKE negotiation nor a trusted-third-party used for the IKE authentication can figure out the session keys generated by IKE.

141. The aforementioned encryption techniques and public key standards would have been well known by one of ordinary skill in the art and creating an encrypted communication channel between two computers using either symmetric or asymmetric encryption would have been in the "toolkit" of any ordinary person of skill in the art circa 2000.

V. IDENTIFICATION OF THE PRIOR ART

A. Exhibits 1009-1011 – The Aventail References

142. Aventail Corporation is a Seattle-based provider of VPN solutions for the enterprise market. It was founded in 1996 and was acquired by SonicWall, Inc. in 2007, which, in turn, was acquired by Dell in 2012.

143. Aventail Extranet Center is a comprehensive VPN software solution that allows organizations to share network resources with external users. It is based on established VPN principles that were well known in the mid 1990's.

144. The "Aventail References" are the following three manuals that describe various aspects of the Aventail Extranet Center software:

a) ***Exhibit 1009: Aventail Connect 3.01/2.51 Administrator's Guide (ACAG);***

b) ***Exhibit 1010: Aventail Connect 3.01/v2.51 User Guide (ACUG); and***

c) ***Exhibit 1011: Aventail Extranet Center v3.0 Administrator's Guide (AECAG).***

145. The documents cross-reference each other, which is logical as they are describing two components of a single system that are designed to work together (i.e., the Aventail Connect client running on the client computer, and the Aventail Extranet Server running on a server computer). *See, e.g.*, Ex. 1009 (ACAG) at 3, 5, 6, 11, 14, 19, 36-40, 44, 46, 48, 50, 52, 55-56, 57, 61-71, 76-77, 79-83, 92-94, 96, 99, 107-109. Given the close interrelationship between the documents, the fact that were distributed together in a single product, and their extensive cross-referencing of the other document, a person of ordinary skill in the art would have understood that the ACAG incorporated by reference the ACUG and AECAG. These manuals describe the operation of the Aventail Extranet Center software and provide detailed guides on its installation configuration, management, and use. The Aventail Connect Administrator's Guide (ACAG) provides a comprehensive description of the operation of the Aventail scheme. In the discussion below, I am relying on the description in the ACAG. I also refer to supporting or corresponding portions of the other two Aventail References as appropriate to further explain how the Aventail scheme functions.

B. Exhibit 1007 – U.S. Patent 6,496,867 to Beser and Borella

146. ***Exhibit 1007 (Beser)*** is a copy of ***United States Patent No. 6,496,867 to Nurettin B. Beser and Michael Borella***.

147. I understand that because Beser was filed on August 27, 1999, several months before the February 2000 effective filing date of the '705 and '009 patents, it is prior art to the claims of the '705 and '009 patents under 35 U.S.C. § 102(e).

C. Request for Comment (RFC) Publications

148. There are a number of publications that I discuss in this report that are "Request For Comments" documents or "RFCs" prepared and distributed under a formalized publication process overseen by one of several Internet standards or governing bodies, such as the Internet Engineering Task Force (IETF) or the World Wide Web Consortium (W3C).

149. The way IETF RFC publications are prepared and released to the public in a formalized and structured process. In fact, the RFC development and publication process itself is described in an RFC – RFC 2026, dated October 1996. That RFC explains that that RFC publications and "Internet-Drafts" are widely disseminated on the Internet. For example, § 2.1 of RFC 2026 explains:

> Each distinct version of an Internet standards-related specification is published as part of the "Request for Comments" (RFC) document series. This archival series is the official publication channel for Internet standards documents and other publications of the IESG, IAB, and Internet community. RFCs can be obtained from a number of Internet hosts using anonymous FTP, gopher, World Wide Web, and other Internet document-retrieval systems.

Ex. 1036 (RFC 2026) at 6.

150. RFC 2026 further explains that once a standard is adopted, it will be formally published. Ex. 1036 (RFC 2026) at 19:

> If a standards action is approved, notification is sent to the RFC Editor and copied to the IETF with instructions to publish the specification as an RFC.

151. Draft IETF standards-track and other documents are also formally published and widely distributed. Ex. 1036 (RFC 2026) at 8:

> During the development of a specification, draft versions of the document are made available for informal review and comment by placing them in the IETF's "Internet-Drafts" directory, which is replicated on a number of Internet hosts. This makes an evolving working document readily available to a wide audience, facilitating the process of review and revision.

152. RFC documents are published on a specific date, which starts a period for others to provide comments on the document. Ex. 1036 (RFC 2026) at 20:

> These minimum periods are intended to ensure adequate opportunity for community review without severely impacting timeliness. These intervals shall be measured from the date of publication of the corresponding RFC(s), or, if the action does not result in RFC

The publication date of each RFC is contained in the RFC, typically in the top right corner of the first page of the document. This is the date it was released for public distribution on the Internet.

153. Each RFC document is uniquely numbered. If it becomes necessary to make a substantive change (*e.g.*, other than a minor typographical error), the RFC will be republished with a different number. Ex. 1036 (RFC 2026) at 21:

```
        the IESG or RFC Editor may be asked to republish the RFC (with
a new number) with corrections
```

154. The formalized process of preparing, publishing and widely distributing RFC documents is a very important part of the Internet culture, which works to develop standards in an open and transparent process. It is also important to the adoption of these standards, and the stability and functionality of the Internet for developers to adhere to standards and evolving "best practices."

155. Because RFC documents are formally prepared and published, and are then widely distributed without any restrictions, they are printed publications as I understand that concept in the patent law.

D. Exhibit 1008 – RFC 2401

156. ***Exhibit 1008*** is a copy of ***IETF Request for Comments 2401: "Security Architecture for the Internet Protocol"*** (***RFC 2401***).

157. I understand that because RFC 2401 was published in November 1998, more than a year before the February 2000 effective filing date of the '705 and '009 patents, it is prior art to the claims of the '705 and '009 patents under 35 U.S.C. § 102(a).

E. Exhibit 1013 – RFC 2543

158. ***Exhibit 1013*** is a copy of ***IETF Request for Comments 2543: "SIP: Session Initiation Protocol"*** (***RFC 2543***).

159. I understand that because RFC 2543 was published in March 1999, before the February 2000 effective filing date of the '705 and '009 patents, it is prior art to the claims of the '705 and '009 patents under 35 U.S.C. § 102(a).

VI. OVERVIEW OF THE PRIOR ART

A. Aventail

160. The Aventail Extranet Center is a client/server software suite for managing (and establishing connections to) extranets. Whereas an "intranet" refers to a internal or private network that is protected at least by physical security and generally can only be accessed by members of an organization, company, etc. that are locally connected to that network, an "extranet" extends some (or all) of the resources of an intranet to users over a wide area network (for example, the Internet). One common use of extranets is for a company to allow customers or clients to access a portion of the local (or private) corporate network remotely over the Internet. Because a corporate network often contains confidential information, most extranets employ security measures in order to prevent access by unauthorized users. IETF RFC 2547 specifically mentions the concepts of extranet and intranet as related to VPNs:

> If all the sites in a VPN are owned by the same enterprise, the VPN is a corporate "intranet". If the various sites in a VPN are owned by different enterprises, the VPN is an "extranet". A site can be in more than one VPN; e.g., in an intranet and several extranets. We regard both intranets and extranets as VPNs. In general, when we use the term VPN we will not be distinguishing between intranets and extranets.

Ex. 1037 (RFC 2547) at 2.

161. As the Aventail References explain, the Aventail Extranet Center software allows a client (for example, a user's laptop) to establish a secure virtual private network connection with a private network, by encrypting the communications between the client and the private network, thereby allowing the client to securely communicate with devices on that private network. *See* Ex. 1009 (ACAG) at 7.

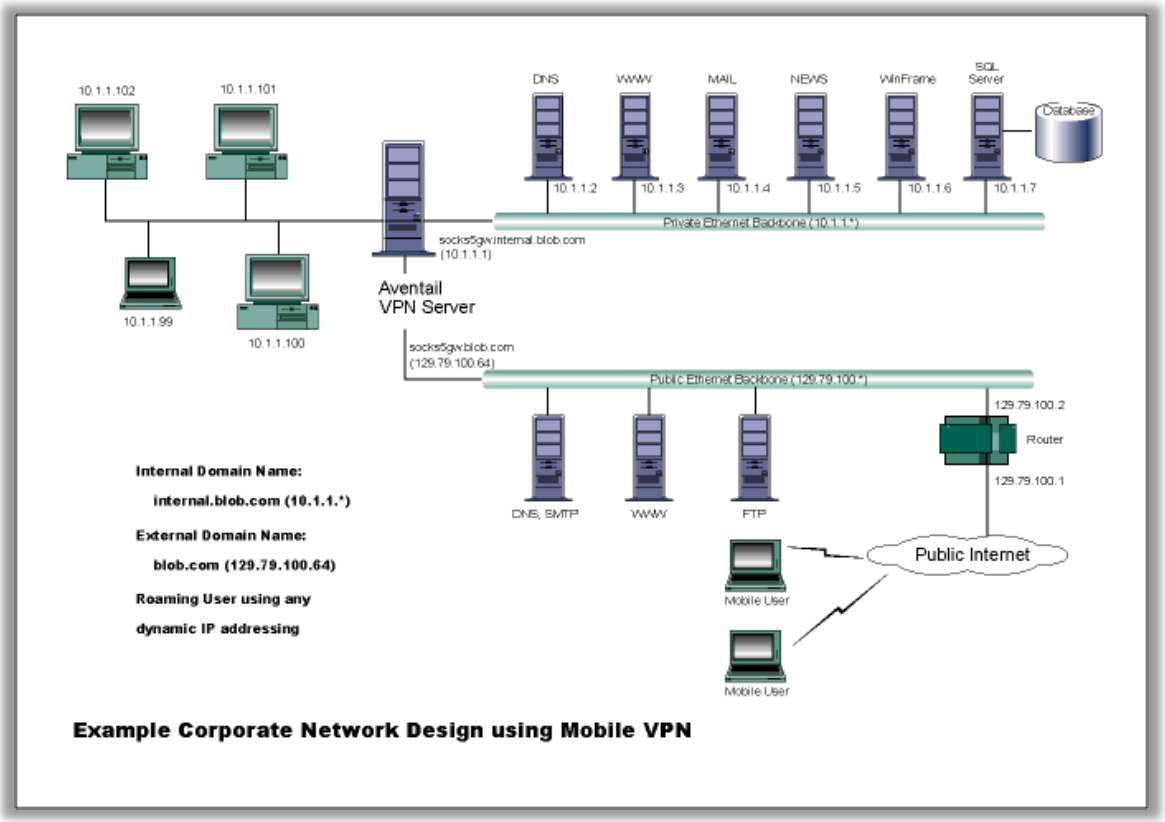
162. Aventail Extranet Center follows the Socket Secure (SOCKS) Protocol Version 5. This protocol was described in IETF RFC 1928 (Ex. 1018) and originated in a 1992 paper (Ex. 1030) by David Koblas and Michelle R. Koblas ("SOCKS," USENIX UNIX Security Symposium III, USENX Association, 1992). This protocol enables a client within an internal network protected by a firewall (e.g., a corporate local area network) to establish a connection to a server on the network external to the firewall (e.g., the internet). A network service running on a SOCKS server is used to as a transient point for the connection in way that is transparent to the client application. The SOCKS server can be the

firewall host itself (as described in the paper by Koblas and Koblas (Ex. 1030)) or a dedicated server deployed on the internal network behind the firewall (as in the more general framework of RFC 1928 (Ex. 1018)).

1. Overview of Aventail Extranet Center

163. Below I provide a high-level overview of the operation of the Aventail Extranet Center software for the establishment of an encrypted VPN connection between a mobile user and a corporate network, omitting other operational scenarios.

164. The Aventail Connect software is installed on the client machine (e.g., a user's desktop or laptop computer). It starts automatically when the machines boots and operates in a way that is transparent to client applications. *Infra* at ¶171. The Aventail ExtraNet Server software is installed on a server that will be typically located behind a firewall of a corporate network, as illustrated in example of the figure below, where Aventail Connect runs on the two laptops labeled "Mobile User" and Aventail ExtraNet Server runs on the server labeled "Aventail VPN Server."



Ex. 1009 (ACAG) at 72.

165. Aventail Connect and Aventail ExtraNet Server work together to establish the encrypted VPN connection. *Infra* at ¶170. When an application on the client computer wishes to communicate over the network, Aventail Connect intercepts this request. *Infra* at ¶¶214-216, 219-254. If the request is for a resource on the corporate network, Aventail Connect redirects it to the ExtraNet Server. *Id.* If the request is for a non-secure resource, Aventail Connect passes the request to the TCP/IP stack of the operating system for normal processing. *Infra* at ¶239.

166. Upon receiving the request, the Aventail ExtraNet Server examines the source of the request and determines the authentication method and associated

credentials (such as Username/Password, token passcode, digital certificate) for that source. *Infra* at ¶241-254. The ExtraNet Server then establishes an encrypted tunnel with Aventail Connect, through which it prompts the user for his or her credentials. *Id.*

167. Aventail Connect will then forward the user's credentials via the encrypted tunnel, and the ExtraNet Server will verify the validity of these credentials. *Infra* at ¶241-254.

168. Following administrator-defined policies and rules, Aventail ExtraNet Center determines whether access to the desired resource on the corporate network should be granted. *Infra* at ¶244. If access is granted, the application on the client machine will be allowed to communicate with the resource, with Aventail ExtraNet Center proxying all connections to secured resources.

169. As outlined above, the central software components of the Aventail Extranet Center are the Aventail Connect client software (which, for example, is executed on one or more computers such as desktop workstations or mobile laptops) and the Aventail Extranet Server software, which runs on a gateway computer (a server) that is connected to both the Internet and to the private network to which access will be provided. I will address each of these software components in turn, starting with Aventail Connect.

2. Aventail Connect

170. The Aventail References describe Aventail Connect as follows:

> 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.

Ex. 1009 (ACAG) at 7.

171. Aventail Connect can be configured to run "transparently" on a user's computer, meaning that the Aventail Connect client would handle examining and intercepting connection requests and establishing connections without requiring the user to take any action:

> 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.

Ex. 1009 (ACAG) at 7.

172. As the Aventail References explain, when the Aventail Connect software is running, it will automatically route network traffic destined for the private network to a proxy server running the Aventail Extranet Server software:

> When you run Aventail Connect on your system, it automatically routes appropriate network traffic from a WinSock application to an extranet (SOCKS) server, or through successive servers. (WinSock is a Windows component that connects a Windows PC to the Internet using TCP/IP.) The SOCKS server then sends the traffic to the Internet or the external network. Network administrators can define a set of rules that route this traffic.

Ex. 1009 (ACAG) at 7.

3. Aventail Extranet Server

173. Aventail explains that Aventail Connect and the Aventail ExtraNet Server work together, with Aventail Connect acting as a proxy client. Ex. 1009 (ACAG) at 7. One of ordinary skill in the art reading this portion of Aventail would understand that the Aventail Extranet Center uses a client/server architecture where multiple client computers running Aventail Connect communicate with the Aventail Extranet Server. Indeed, the Aventail Extranet Center Administrator's Guide confirms that this is how Aventail operates:

> Aventail ExtraNet Server is the server component of the Aventail ExtraNet Center, a client/server solution for management of sophisticated extranets. Setup of the Aventail ExtraNet Center requires that installation on both a server and multiple client machines.

Ex. 1011 (AECAG) at 5.

174. Aventail explains that the Aventail ExtraNet Server is a server that implements the SOCKS v5 protocol. Ex. 1009 (ACAG) at 7 ("Aventail Connect works with the Aventail ExtraNet Server, the SOCKS 5 server component of the Aventail ExtraNet Center."). One of ordinary skill in the art reading that sentence

would understand that the Aventail Extranet Server works in conjunction with client machines running the Aventail Connect software to authenticate users and manage network traffic from and to a private network. Indeed, the Aventail ExtraNet Center Administrator's Guide explains that the ExtraNet Server performs just this function:

> **Aventail ExtraNet Server:** The primary component of Aventail ExtraNet Center is the ExtraNet Server. This is a SOCKS v5 proxy server that manages the authentication of users and processes all of the connection requests. Aventail ExtraNet Server can manage traffic for both incoming (external users attempting to reach internal network resources) and outgoing (internal users attempting to reach external network resources) network traffic.

Ex. 1011 (AECAG) at 5.

175. One of ordinary skill in the art reading in Aventail that the Aventail ExtraNet Server is a SOCKS 5 server (EX. 1009 (ACAG) at 7), would understand that the Aventail Extranet Server provides SOCKS network proxy functionality by implementing the enforcement of various access control and authentication policies on users, groups, encryption methods, networks, connection endpoints, and traffic routing. Indeed, the Aventail ExtraNet Center Administrator's Guide explains that the Aventail ExtraNet Server enforces such policies, which can take into account temporal constraints (e.g., rules applicable on certain days and at certain hours). Ex. 1011 (AECAG) at 9-11. The Aventail ExtraNet Server Administrator's Guide

provides further detail on how these policies can be configured using access control rules and filters, and provides the following example:

> Once you select the XYZ network as a source network, you can then refine the access rights of anyone coming from that network, e.g., by limiting access to only a specific destination network (Destination Networks tab), during certain times of the day (Advanced tab | Time Editor).

Ex. 1011 (AECAG) at 22.

176. Specifically, there are four different types of policies that the Aventail Extranet Server can enforce. Ex. 1011 (AECAG) at 9:

> There are four types of rules that can be used to build an Aventail ExtraNet Server policy

177. First, there are Access Control Rules. These rules can be used to specify who can access which networks resources (i.e., individual computers on a network or services on the network such as e-mail or FTP servers) as well as any conditions required for access (i.e., an encrypted connection). See Ex. 1011 (AECAG) at 9:

> Access control rules define the network resources and services that are accessible to users and groups based on where they are coming from, what day or time it is, how they authenticated, and what their encryption strength is.

178. Second, the Aventail Extranet Server provides for Authentication Rules. These rules provide a mechanism to specify the different ways a user is allowed to authenticate with the Server based on the network location of the user. Ex. 1011 (AECAG) at 9 ("Authentication rules define the authentication options

available to users or groups based on where they are coming from."). For example, a user who is on the same private network as the Server may just provide a cleartext password, whereas a remote user may be required to provide challenge-response credentials using an encrypted connection (such as SSL).

179. Third, Filter Rules are provided, offering control over the type of HTTP content (website content) that the Server allows users to access, as well as a forwarding authentication information if possible. Aventail describes these two filters as follows:

> ***HTTP Content Filter***
>
> The HTTP Content filter offers both fine-grained, text-editable content control (Aventailfilter), as well as a predefined definition file (SmartFilter) of "deny" categories. Usually, you will use the text-editable filter only when content gets past the SmartFilter and continues to the end user.
>
> ***HTTP Authentication Forwarding filter***
>
> The HTTP Authentication Forwarding filter allows you to forward user credentials in the HTTP header to an IIS Web server or any Web server that supports the same protocol(s) used by the Aventail ExtraNet server. This eliminates the need to repeatedly enter authentication information.

Ex. 1011 (AECAG) at 48.

180. Fourth, Proxy Chaining Rules are available to allow the Aventail Extranet Server to provide access to other private networks that are themselves behind one or more other Aventail Extranet Servers. Ex. 1011 (AECAG) at 10 ("Defines the methods for accessing network destinations located behind other ExtraNet Servers.").

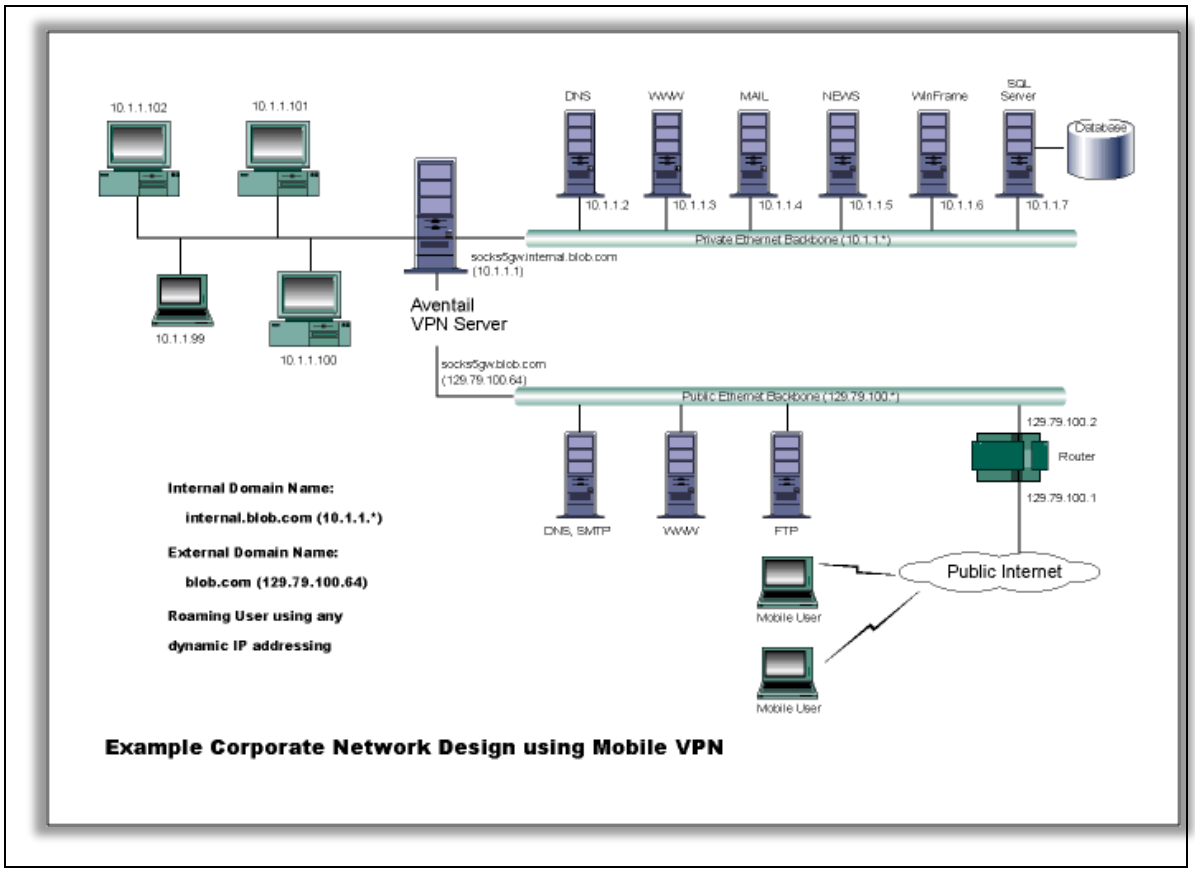
181. The various types of policy rules described above are stored and implemented by the Aventail Extranet Server, and can be examined or modified by an administrator through a piece of software called the Aventail Policy Console, as Aventail explains:

> **Aventail Policy Console:** The Aventail Policy Console is the graphical administrative tool for creating, viewing and managing the policies for your extranet. It can also be used for starting and stopping the ExtraNet Server as well as viewing log and license files.
>
> The Policy Console provides a graphical front-end for the configuration file that the Aventail ExtraNet Server uses. The Policy Console can be run locally on the machine that the ExtraNet Server is installed on or remotely to manage a server that resides on another machine. When the Policy Console is being run remotely, it will establish a secure LAN, WAN or Internet connection via the Management Server (see below). A remote Policy Console running on a Windows NT machine can configure a UNIX Aventail ExtraNet Server and vice versa.

Ex. 1011 (AECAG) at 5.

4. Aventail Extranet VPN

182. Having provided a basic overview of Aventail Connect and the Aventail Extranet Server, I will now discuss how Aventail describes these pieces of software working together to create a secure VPN extranet connection for remote clients. To do this, I will make again reference to the figure on page 72 of ACAG that illustrates one possible network configuration for the Aventail Extranet Center software:



Ex. 1009 (ACAG) at 72.

183. Aventail explains that this picture shows a corporate network with both a private network segment (addresses in the subnet 10.1.1.*) and a public network segment (addresses in the subnet 129.79.100.*). The hosts on the external segment (e.g., the WWW host) can be accessed by all external users. The Aventail VPN Server has network interfaces on both the internal and external networks (internal IP address 10.1.1.1 and external IP address 129.79.100.64). It runs the Aventail Extranet Server to provide an encrypted virtual connection to private network hosts (e.g., the SQL server) to the external users running the Aventail Connect Software:

> 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.
>
> . . .
>
> 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.

Ex. 1009 (ACAG) at 72-73.

184. The above figure (Ex. 1009 (ACAG) at 72) shows two mobile users (running the Aventail Connect software) that are connected to the Internet. Without the Aventail software, these users would be able to connect only to the servers on the public network segment (labeled "Public Ethernet Backbone"): the "DNS, SMTP" server, the WWW server, and the FTP server shown in the middle

of the figure. However, these users would not be able to access the hosts on the private network segment (labeled "Public Ethernet Backbone"): the DNS server, the WWW server, the MAIL server, the NEWS server, the WinFrame server, and the SQL server, shown at the top of the figure.

185. Using the Aventail Connect software, however, the mobile users can connect to the Aventail VPN server (running the Aventail Extranet software) which gives these mobile users access to both the private workstations pictured at the top left of the figure and the private servers pictured at the top right of the figure. In particular, the VPN connection established by the Aventail Connect software on the mobile user machine and the Aventail Extranet Server software on the Aventail VPN server host would allow the mobile users to access, among other services, a mail server on the private network, a news server on the private network, and a database (SQL) server on the private network.

186. Using standard networking terminology, the client computer running Aventail Connect is termed a "proxy client" while the Aventail VPN server is called a "proxy server." Indeed, in the field of networking, the term "proxy server" is used to describe a computer that acts as an intermediary on behalf of one or more other servers; a proxy "client" can connect to this proxy server, and the proxy server will then be responsible for relaying messages between the client and the

other servers/services to which the proxy server offers access. *See, e.g.*, Ex. 1019 (Microsoft Computer Dictionary) at 363 (definition of "proxy server").

187. A proxy server can provide anonymity, because the servers with which the proxy server communicates need not be informed of the identity of the proxy client. *See, e.g.*, Ex. 1019 (Microsoft Computer Dictionary) at 363 (definition of "proxy"). A proxy server can also be used to prevent a proxy client from accessing certain servers, content, or services, by refusing to provide such access to the client. For example, a company may wish to only allow users on its network to access certain types of content outside of that network (e.g., web servers but not mail servers) or only to have access at certain hours during the day. If the proxy client is required to go through the proxy server in order to gain outside network access, then access can be restricted in this way according to the policies of the company. *See, e.g.*, Ex. 1011 (AECAG) at 48-52 (discussing content filtering).

188. In the picture above (Ex. 1009 (ACAG) at 72), the Aventail Extranet Server acts as a gateway, sitting between the public internet and the private corporate network, passing network traffic (data packets) between the two:

> 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.

Ex. 1009 (ACAG) at 72.

5. Types of Communication Supported by Aventail

189. Aventail teaches that its technology can be used for a variety of purposes. For example, Aventail explains that its networking scheme supports end-to-end encryption techniques such as SSL in a "protocol-independent fashion". Ex. 1009 (ACAG) at 47. Similarly, Aventail discloses configurations in which various network-based services are provided by way of encrypted network communication channels. *See, e.g.*, Ex. 1009 (ACAG) at 72 (diagram showing VPN with DNS, WWW, mail, news, WinFrame, and SQL servers). DNS, WWW, mail, news, WinFrame, and SQL servers are shown in the diagram as different services because they generally operate using different protocols and are accessed with different application programs. For example, it was well known circa 2000 that an email application could be used to access an email server using an email protocol, while a news application could be used to access a news server using a news protocol. Aventail discusses web browser applications, which were known to use the HTTP protocol. *See, e.g.*, Ex. 1009 (ACAG) at 65. Aventail also discusses Extranet Neighborhood application that uses the SMB and NetBIOS over TCP protocols. *See, e.g.*, Ex. 1009 (ACAG) at 1, 91. Aventail thus discloses an encrypted channel that supports at least the following services: WWW server, HTTP protocol, web browser application, email application, mail protocol, mail

server, WinFrame server, SQL server, SMB protocol, NetBIOS over TCP protocol, and Extranet Neighborhood application.

6. Physical Implementation of the Networks Used by Aventail

190. The Aventail scheme fundamentally involves networks, but Aventail does not explain in detail the lower level implementation of the networks. Aventail does explain that a modem-based network can be used. *See, e.g.,* Ex. 1009 (ACAG) at 52. Though Aventail does not explicitly explain this, persons of skill in the art would have understood that network channels implemented with modems are modulated by varying a carrier signal. The Microsoft Computer Dictionary (4th ed.), for example, explains:

> When transmitting, modems impose (modulate) a computer's digital signals onto a continuous carrier frequency on the telephone line. When receiving, modems sift out (demodulate) the information from the carrier and transfer it in digital form to the computer.

Ex. 1019 (Microsoft Computer Dictionary) at 294.

191. Aventail also shows that Ethernet can be used as the lower level implementation of its network, for both the public and private portions. *See, e.g.,* 1009 (ACAG) at 72.

192. Persons of ordinary skill in the art would have understood that a variety of other network types could also have been used to implement the Aventail systems.

7. Security and Encryption in Aventail

193. The Aventail system provides security policies that can be configured by an administrator to require that a user outside the private network authenticate and/or establish a secure connection in order to access the private network. *See* Ex. 1009 (ACAG) at 73:

> 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.

and Ex. 1009 (ACAG) at 12:

> 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.

194. For example, the Aventail Extranet Server will only proxy traffic from a client that is authorized to communicate with a particular computer or service on 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.

Ex. 1009 (ACAG) at 77.

195. Aventail explains that users have to supply valid authentication credentials in order to be authenticated by the proxy server (AES). *See* Ex. 1009 (ACAG) at 7 ("…users will interact with Aventail Connect only when it prompts them to **enter authentication credentials** for a connection to **a secure extranet (SOCKS) server**…") (emphasis added); *id*. at 46-62. The authentication request is in the form of a message or a communication between the proxy server and the client.

a) SOCKS v5

196. Aventail explains that many of the security features provided by the Aventail Extranet Server are derived from the Aventail Extranet Server's implementation of the SOCKS v5 protocol. Ex. 1009 (ACAG) at 7 ("Aventail connect works with the Aventail ExtraNet Server, the SOCKS 5 server component of the Aventail ExtraNet Center.").

197. SOCKS stands for Sockets Secure and is an Internet protocol that specifies how data packets should be routed between a client and server using a proxy. Ex. 1018 (RFC 1928 – "SOCKS Protocol Version 5"). One of ordinary skill in the art would have been well versed in the SOCKS protocol. Whereas earlier versions of the SOCKS protocol did not provide many security features,

SOCKS v5 added substantial security functionality, such as authentication and encryption. As Aventail explains:

> Enter SOCKS v5, an Internet Engineering Task Force (IETF)-approved security protocol targeted at securely traversing corporate firewalls. SOCKS was originally developed in 1990, and is now maintained by NEC. SOCKS acts as a circuit-level proxy mechanism that manages the flow and security of data traffic to and from your local area network (LAN) or extranet. An application whose traffic is proxied by SOCKS is considered "socksified." SOCKS is more than a standard security firewall. Other features:
>
> - Client Authentication: (SOCKS v5 only) Authentication allows network managers to provide selected user access to internal and external areas of a network.
> - Traffic Encryption: (SOCKS v5 only) Encryption ensures that network traffic is private and secure.
> - UDP Support: (SOCKS v5 only) User Datagram Protocol (UDP) traffic has traditionally been difficult to proxy, with the exception of SOCKS v5.
> - Aventail Connect supports X.509 client certificates within SSL: Includes a Certificate wizard for generating and processing client certificate requests.
> - Cross-Platform Support: Unlike many other security solutions, SOCKS can be used on various platforms, such as Windows NT, Windows 95, Windows 98, and various forms of UNIX.

Ex. 1009 (ACAG) at 6-7.

198. As Aventail explains, a SOCKS 5 server (such as the Aventail Extranet Server) could require a user to authenticate with the server before the user was allowed to access the resources provided by that server (in this case, computers, data, and services available on the private network that sits behind the Aventail Extranet Server). Ex. 1009 (ACAG) at 42 ("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.").

b) Authentication Modules

199. Aventail states that one step in configuring Aventail Connect is to specify which authentication modules Aventail Connect should make available for use: SSL, CRAM, CHAP, UN/PW, SOCKS v4, or HTTP Basic (username/password). Ex. 1009 (ACAG) at 27 ("**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)."). I will address each of the authentication modules in turn.

200. CRAM stands for Challenge Response Authentication Method and is used to send authentication information (i.e., username and password) over the network using an SSL encrypted connection if possible. Ex. 1009 (ACAG) at 27 ("**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.").

201. CHAP stands for the Challenge Handshake Authentication Protocol and provides a mechanism for sending a username and password over the network

in an encrypted format. Ex. 1009 (ACAG) at 28 ("CHAP: The Challenge Handshake Authentication Protocol (CHAP) sends your username and password encrypted across the network to the destination server.").

202. UN/PW stands for username/password and is a mechanism for sending a username and password across a network for authentication. Ex. 1009 (ACAG) at 28 ("**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.").

203. SOCKS v4 – the precursor to SOCKS v5 – lacked many of the security facilities of SOCKS v5 but did allow a user to send a username (but no password) with a connection request. Ex. 1009 (ACAG) at 28 ("**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.").

204. Another option was to allow for a username/password to be sent over HTTP. Ex. 1009 (ACAG) at 28 ("**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.").

205. SSL stands for Secure Sockets Layer and is a protocol for creating an encrypted communication link. Ex. 1009 (ACAG) at 27 ("**Secure Sockets Layer**: Secure Sockets Layer (SSL) is a session-layer protocol for securing connections in a general, protocol-independent manner."). SSL servers use encryption and digital signatures based on a public-key cryptosystem to establish an encrypted communication session:

> 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).

Ex. 1009 (ACAG) at 47.

206. As Aventail goes on to explain, in order to verify the identity of the server providing a public key, digital certificates issued by a trusted certificate authority (see ¶136) are used to avoid impersonation:

> 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.

Ex. 1009 (ACAG) at 47.

8. Aventail Extranet Center – Operation

207. The Aventail Extranet Center software consists of a protocol executed by the Aventail Connect software running on a client device, and the Aventail Extranet Server software running on a VPN server device.

208. Before I discuss how Aventail Connect operates to create an encrypted channel with the Aventail Extranet Server (and, ultimately, a secure connection to the VPN), it is first important to understand the flow of network

communications in a scenario in which Aventail Connect is not running – that is, how a connection is made between an application that wishes to communicate with a remote host using the standard network components of an operating system, such as Microsoft Windows or Linux.

209. First, I note that one of ordinary skill in the art would understand that the process of communicating with a remote host starts with an application requesting to lookup the network address associated with a domain name. For example, if a user enters "www.yahoo.com" into a web-browser application (running on a client computer that is also running Aventail Connect), the web-browser will send a request to lookup the network address associated with "www.yahoo.com." The returned network address is then used to establish a network connection with the computer identified as "www.yahoo.com."

210. Aventail explains that there are three basic steps that a client computer must perform to communicate with a remote host: 1) obtain the IP address of the remote host; 2) request a connection to the remote host; and 3) send and receive data from the remote host once a connection is obtained. Aventail describes these steps in detail, referring for concreteness to the Windows operating system:

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

Ex. 1009 (ACAG) at 8.

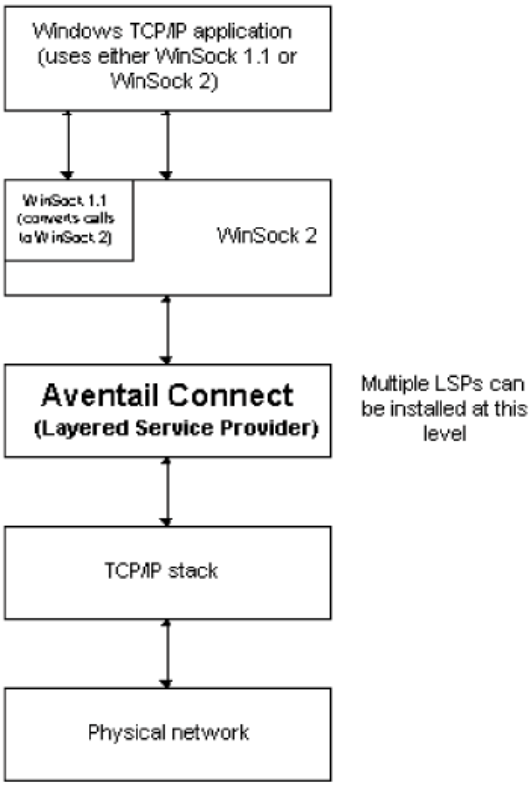
211. As I have previously explained in Section IV.C, the DNS system provides a mechanism that enables a client to send a query that contains a hostname (a textual identifier such as "www.example.com") to a DNS name server to look up the network address (the IP address) of the computer associated with the hostname.

212. And as I have explained in Section IV.B, TCP/IP is the name for a set of protocols that are used to address and route traffic over the Internet. The various components of a TCP/IP implementation are generally referred to as the TCP/IP protocol "stack." "WinSock" – which is short for the Windows Socket API – is the software library provided by the Windows operating system to allow applications to interface with the components the TCP/IP protocol stack.

213. I will next discuss how the three steps detailed above in Paragraph 210 are modified when the Aventail Connect software is running on a client computer.

214. When Aventail Connect is running, it inserts itself as an intermediary "layer" in the computer's networking stack and intercepts traffic from TCP/IP applications:

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



Ex. 1009 (ACAG) at 9.

215. Because the Aventail Connect software sits between the application running on the computer and the TCP/IP stack, Aventail Connect can evaluate, route, and encrypt any communications that make use of TCP/IP. See Ex. 1009 (ACAG) at 10:

> 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).

216. Aventail Connect can therefore work with (any application running on Windows that makes use of the TCP/IP protocol, such as web browsers and email programs. *See, e.g.*, Ex. 1009 (ACAG) at 8:

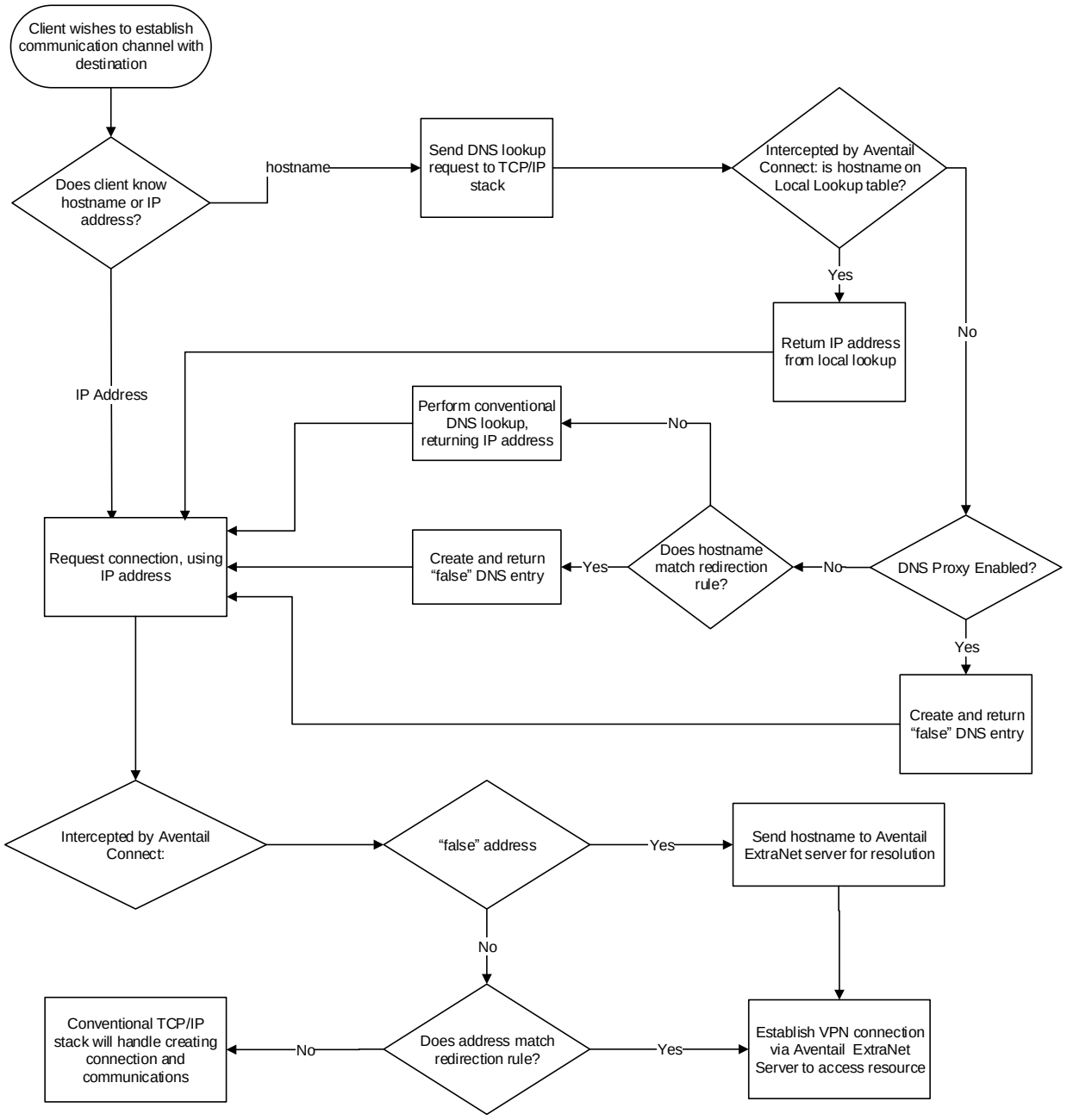
> Windows TCP/IP networking applications (such as telnet, e-mail, Web browsers, and ftp) use WinSock (Windows Sockets) to gain access to networks or the Internet. WinSock is the core component of TCP/IP under Windows, and is the interface that most Windows applications use to communicate to TCP/IP.

See also Ex. 1031 (Windows NT for Dummies) at 14.

217. As Aventail explains, the same three basic steps described above are still performed when Aventail Connect is running, but Aventail Connect modifies each of these steps in order to proxy certain network traffic to computers on a private network through the Aventail Extranet Server. See Ex. 1009 (ACAG) at 11:

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.

218. I explain each of these steps in detail below, but first provide a flowchart that I prepared illustrating these steps:



a) Step 1 – DNS Query Interception

219. With respect to the first step, Aventail explains that the application will send a DNS lookup query that the Aventail Connect software will intercept. Ex. 1009 (ACAG) at 11. Of course, if the application already knows the IP address of the device with which it wishes to communicate, no DNS lookup is necessary and this step is skipped:

> 1. The application does a DNS lookup to convert the hostname to an IP address. If the application already knows the IP address, this entire step is skipped.

Ex. 1009 (ACAG) at 11.

220. When a DNS query is sent, Aventail Connect will intercept the query, and will first check the received domain name against a table of "local domain strings" and then against a table of "redirection rules." Ex. 1009 (ACAG) at 11:

> 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.

I discuss each of the topics in detail below.

(i) **Local Domain Strings**

221. The "local domain strings" referenced above are part Aventail's Local Name Resolution functionality, which is an optional feature of the Aventail system that allows hostname resolution without querying a remote name server (i.e., without making a query over the Internet). See Ex. 1009 (ACAG) at 41:

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

222. A "local domain string" (or "local name") is a domain name (hostname) value (e.g., "aventail.com") that is stored in a local domain table on the computer running Aventail Connect. See Ex. 1009 (ACAG) at 41:

> 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.

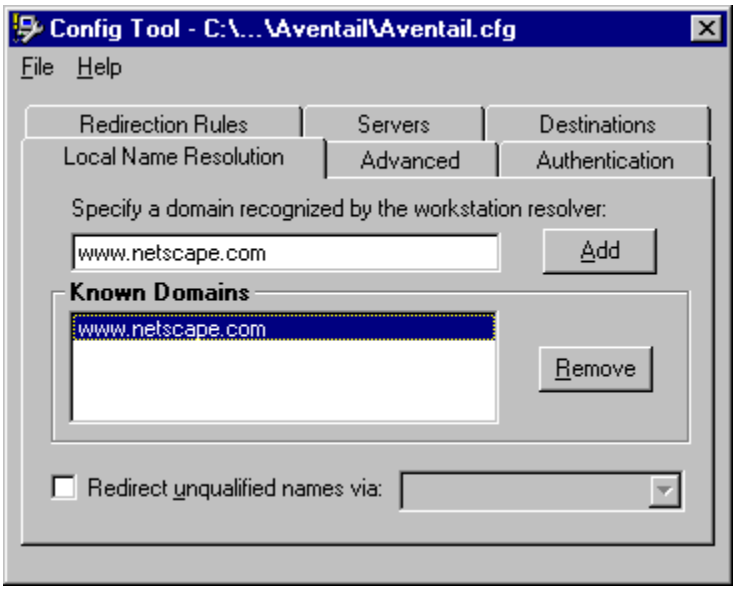
223. As Aventail explains, the Local Name Resolution function is essentially just a shortcut that allows a domain name (hostname) matching a defined string to be evaluated locally, rather than proxied through to the Aventail Extranet Server for resolution:

> 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.

Ex. 1009 (ACAG) at 41.

224. While Aventail does not elaborate further on this functionality, one of ordinary skill in the art would understand that the domain name strings stored in the local name resolution table would include both hostnames that are only valid on the local network or a remote private network (i.e., hostnames that cannot be

resolved using conventional DNS lookups, such as local computer names or workgroup names), as well as domain names that can be resolved using conventional DNS lookups (such as public websites like aventail.com and netscape.com) for which an IP address may have been cached or otherwise stored on the local machine. See Ex. 1009 (ACAG) at 41. Aventail shows a screenshot of a configuration program ("Config Tool") in which the "non-secure" domain name "www.netscape.com" has been added to the Local Name Resolution list:



Ex. 1009 (ACAG) at 41.

(ii) **DNS Proxy**

225. As I have described above, Aventail explains that if the domain name in a request does not match a local domain string (or if there are no local domains stored at the client), the domain name value in the request will then be compared

against a second table of values, called redirection rules, unless the "DNS proxy option" is enabled. Ex. 1009 (ACAG) at 11-12.

226. If the DNS proxy option is enabled, then the proxy server will be responsible for resolving the DNS lookup request, so Aventail Connect will create a "false" DNS entry that it can later recognize and return this entry to calling application:

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

Ex. 1009 (ACAG) at 12.

227. Thus, when DNS Proxy functionality is enabled, Aventail Connect will route ***all*** DNS requests not matching a local domain string to an Aventail Extranet server for interception and resolution.

228. Therefore, when DNS Proxy is enabled, an Aventail Extranet Server – as opposed to Aventail Connect – would be responsible for resolving the DNS request and returning an IP address to the client running Aventail Connect. Ex. 1009 (ACAG) at 11-12; *id.* at 61-64.

(iii) **Redirection Rules**

229. If the domain name in the DNS request does not match a "local domain string" and the DNS Proxy option is not enabled, then Aventail Connect will check to see if domain name in the DNS request matches a domain name specified in a "redirection rule." Ex. 1009 (ACAG) at 11-12.

230. In Aventail, redirection rules are stored as a table of values that associate a domain name with a redirection destination for that domain name, along with information specifying how requests containing that domain name are to be handled. Ex. 1009 (ACAG) at 38-39.

231. Thus, redirection rules can be used to specify the hosts/domains for which Aventail Connect will provide access via VPN by proxying through the Aventail Extranet Server. Ex. 1009 (ACAG) at 38:

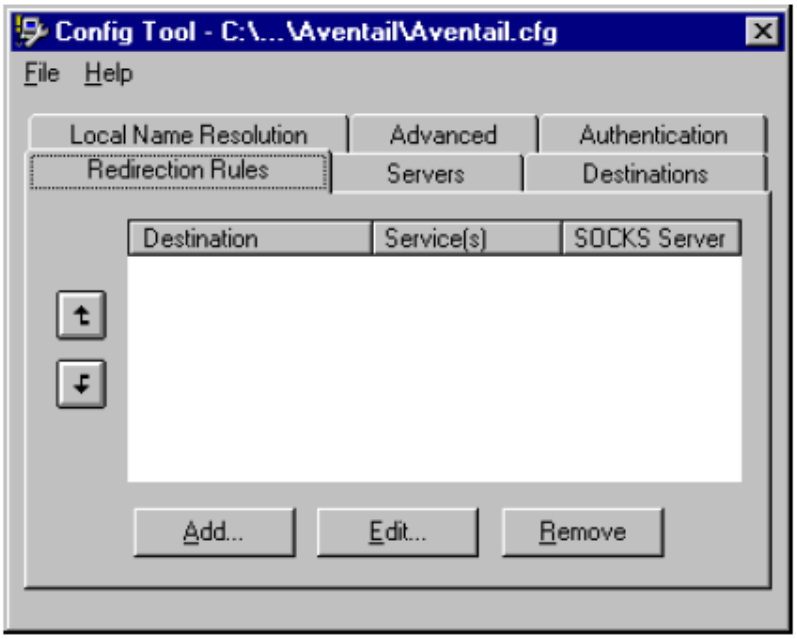
> Once servers and destinations are defined, you can specify how you want Aventail Connect to redirect (or deny) access to various hosts and services such as e-mail, FTP, and HTTP.
>
> Redirection rules are specified on the **Redirection Rules** tab in the Config Tool.

232. Aventail explains that redirection rules are stored on the client computer running Aventail Connect as part of Aventail Connect's configuration file. Ex. 1009 (ACAG) at 15:

> 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.

233. Aventail also explains that redirection rules may be created by a user, or may be established by an administrator using a configuration file used during installation of the Aventail Connect client. ACAG 19-20, 30-41.

234. Aventail provides the following pictures that illustrate the redirection rule table and provide a description of what the fields in the table mean:



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

Ex. 1009 (ACAG) at 38.

235. A redirection rule can specify an individual computer, a network, or a subnetwork as the "Destination" to which traffic should be redirected. *See, e.g.*,

Ex. 1009 (ACAG) at 36-37. A destination can be specified either by using a hostname (or domain name), an IP address, or both. *Id.* at 37 ("Under 'Single host,' type the actual name of the host system and/or its full, numeric IP address."). The options for destinations are illustrated in this table (below):

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

Ex. 1009 (ACAG) at 37.

236. As part of the redirection rule, Aventail provides three options for handling requests, as illustrated in the figure below: all traffic to a particular destination can be blocked (denied); all traffic could be routed to a specified Aventail ExtraNet Server; or traffic can be routed directly to specified destination, bypassing the Extranet Server in its entirety. Ex. 1009 (ACAG) at 40. These

options are mutually exclusive. Ex. 1009 (ACAG) at 40 ("Under 'Proxy Redirection,' select one of three redirection options.").

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.

237. As I have described above, if Aventail Connect determines that the hostname in the intercepted DNS request matches one of the destinations for which a redirection rule has been defined, then Aventail Connect will evaluate the redirection rule to determine if the target host is one for which proxy redirection (and an encrypted communication) through the Aventail Extranet Server is required. Ex. 1009 (ACAG) at 11. If it is determined that such redirection is required, then Aventail Connect will create a "false" DNS entry that it will be able to recognize later when a connection request is sent. *Id.* ("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.").

b) Step 2 – Connection Interception and Setup

238. Once the DNS query has been resolved (step 1) – in other words, once the application has an IP address for the host it wishes to communicate with – the application will send a connection request to the TCP/IP stack, which the Aventail Client will intercept and examine:

> 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.

Ex. 1009 (ACAG) at 12 (initial task of Step 2 and Step 2.a).

239. Based on the examination of the connection request, Aventail explains that Aventail Connect will take one of three actions:

> - 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.

Ex. 1009 (ACAG) at 12, Step 2.a.

240. This first scenario could be the result of either the "DNS proxy option" being selected, or Aventail Connect having determined that the hostname in the DNS request matched a redirection rule. Ex. 1009 (ACAG) at 11-12. The second scenario could result from: 1) a local domain string lookup where the resulting IP address matched a redirection rule; 2) a standard DNS lookup where the resulting IP address matched a redirection rule; 3) a DNS proxy lookup that returned an IP address that matched a redirection rule.

241. In either of these two scenarios, having determined that communications over this connection are to be proxied, Aventail Connect then determines whether this connection should be encrypted. Ex. 1009 (ACAG) at 10 ("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)").

242. To make this determination, Aventail Connect communicates with the Aventail Extranet Server to negotiate how to establish a connection to the destination. More specifically, Aventail Connect interacts with the proxy server (Aventail Extranet Server) to establish an encrypted channel (if necessary) and to perform authentication. As Aventail explains in Ex. 1009 (ACAG) at 12, Step 2.b (initial task and first two substeps):

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.

243. A person of ordinary skill in reading Aventail's discussion of a SOCKS negotiation would have understood that that the negotiation was a defined process specified by a standard. In particular, the SOCKS 5 standard defines SOCKS and specifies the SOCKS negotiation. *See* Ex. 1018 (RFC 1928). According to the SOCKS standard, if the client computer is allowed access to the requested remote host, the SOCKS server will send a "succeeded" response to the client that provides the network address and network port of the server to which the client computer should send its encrypted communications.

6. Replies

The SOCKS request information is sent by the client as soon as it has established a connection to the SOCKS server, and completed the authentication negotiations. The server evaluates the request, and returns a reply formed as follows:

VER	REP	RSV	ATYP	BND.ADDR	BND.PORT
1	1	X'00'	1	Variable	2

Where:

- VER protocol version: X'05'
- REP Reply field:
  - X'00' succeeded
  - X'01' general SOCKS server failure
  - X'02' connection not allowed by ruleset
  - X'03' Network unreachable
  - X'04' Host unreachable
  - X'05' Connection refused
  - X'06' TTL expired
  - X'07' Command not supported
  - X'08' Address type not supported
  - X'09' to X'FF' unassigned
- RSV RESERVED
- ATYP address type of following address
  - IP V4 address: X'01'
  - DOMAINNAME: X'03'
  - IP V6 address: X'04'
- BND.ADDR server bound address
- BND.PORT server bound port in network octet order

Ex. 1018 (RFC 1928) at 5-6.

244. One of ordinary skill in the art reading the Aventail Connect Administrator's Guide would understand that the Aventail Connect client sends a list of authentication methods to the Aventail Extranet Server and the server is responsible for choosing which authentication method to use. Ex. 1009 (ACAG) at

12. The Aventail ExtraNet Center Administrator's Guide confirms that this is how Aventail Extranet Center operated, explaining that:

> When the server receives an incoming client request, it will poll its suite of authentication rules in descending order to match the user's source (IP, host name, domain name, range, or subnet) to a source in an authentication rule. When the server finds a match, it then compares authentication methods available to the client source against methods the client offers the server. The server then selects the appropriate method for the client if it is in the Config file.
>
> - If the source of the client request is "Anywhere," the server will match this rule and look no further.
> - If the server cannot match a client source, it will deny the connection.
> - If the client cannot use any source the server specifies in an authentication rule, the server will deny the connection.

Ex. 1011 (AECAG) at 36. I note that, in the above description, it is understood that the term "server" refers to the Aventail Extranet Server software.

245. As previously discussed, one of the authentication modules that Aventail supports is SSL, and if this module is enabled for a source, then an encrypted channel will be created for Aventail Connect to communicate over the VPN. Again, the Aventail ExtraNet Center Administrator's Guide confirms that this is how the Aventail ExtraNet Center operated, explaining:

> As an example (see above), rule #1 requires a user coming from the company's internal subnet to authenticate with a username and password. Rule #2 requires the user to subauthenticate with username/password; i.e., if the client comes from anywhere other than the internal subnet, it must first establish an SSL (encrypted) session, then authenticate with, in this case, username/password as a secondary method.

Ex. 1011 (AECAG) at 36.

246. Thus, the Aventail Extranet Server will indicate to the Aventail Connect client whether an encrypted communication channel is supported (and

required) for communication with the target device for which the Aventail Extranet Server will act as a proxy.

247. If it is determined that Aventail Connect must use SSL to authenticate and establish a connection to the target device, then Aventail Connect and the Aventail Extranet Server will exchange various messages in order to establish the parameters of the encrypted connection, including public key information, digital certificates, compression options, and cipher options, among other things. *See generally* Ex. 1009 (ACAG) at 46-52.

248. For example, Aventail explains that the Aventail Connect SSL module is responsible for receiving certificate information from the Aventail Extranet Server. *See* Ex. 1009 (ACAG) at 47:

> 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.

Aventail goes on to provide a detailed description of how the SSL module in Aventail Connect should handle certificates received from the Aventail Extranet Server, as illustrated by the screenshot and table below in Ex. 1009 (ACAG) at 48:

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.

249. As can be seen from these illustrations (and the accompanying text on pages 48-49 of ACAG), Aventail Connect is responsible for evaluating these SSL certificates received from a server and taking the action specified.

250. Aventail further explains that, to create the SSL connection, the client and server exchange information in order to determine which encryption cipher to use. Specifically, Aventail states that:

> 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.

Ex. 1009 (ACAG) at 51.

251. After it has been determined whether an encrypted channel is necessary (and, if so, after the encrypted channel has been established), and authentication has been completed, Aventail Connect performs the following action:

> 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.

Ex. 1009 (ACAG) at 12 (third substep of Step 2b).

252. By sending this proxy request to the Aventail Extranet Server, Aventail Connect is instructing the Aventail Extranet Server to proxy Aventail Connect's connection to the target computer on the private network. Ex. 1009 (ACAG) at 60 specifically says:

> 4. Aventail Connect instructs the Aventail ExtraNet Server to proxy its connection to the final destination.

253. Finally, now that the SOCKS negotiation has been completed and a connection has been established, Aventail Connect notifies the application

> c. When the SOCKS negotiation is completed, Aventail Connect notifies the application.

Ex. 1009 (ACAG) at 12, Step 2.c.

254. Note that the execution of the above protocol between Aventail Connect and the Aventail Extranet server is transparent to the application, whose operation begins with a virtual TCP/IP handshake with the remote computer on the private network.

> From the application's point of view, the entire SOCKS negotiation, including the authentication negotiation, is merely the TCP handshaking.

Ex. 1009 (ACAG) at 12, Step 2.c.

c) Step 3 – Encrypted Channel Communications

255. As Aventail explains, once the connection has been set up, the application can send and receive data with the remote computer over the communication channel established by the connection. Ex. 1009 (ACAG) at 12, Step 3 ("The application transmits and receives data.").

256. If it has been determined that the Aventail Extranet Server and the Aventail Connect client both support encryption and that an SSL connection is required for the Aventail Connect client to connect to the VPN, then the communication link between the client and the target device on the private network is a secure, encrypted link:

> 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.

Ex. 1009 (ACAG) at 12.

9. MultiProxy

257. Aventail Connect offers another feature called "Multiproxy", which allows network traffic to be routed through multiple firewalls or proxies, rather than just through a single Aventail Extranet Server. Ex. 1009 (ACAG) at 59-67.

258. The "Aventail Multiproxy" feature is described as:

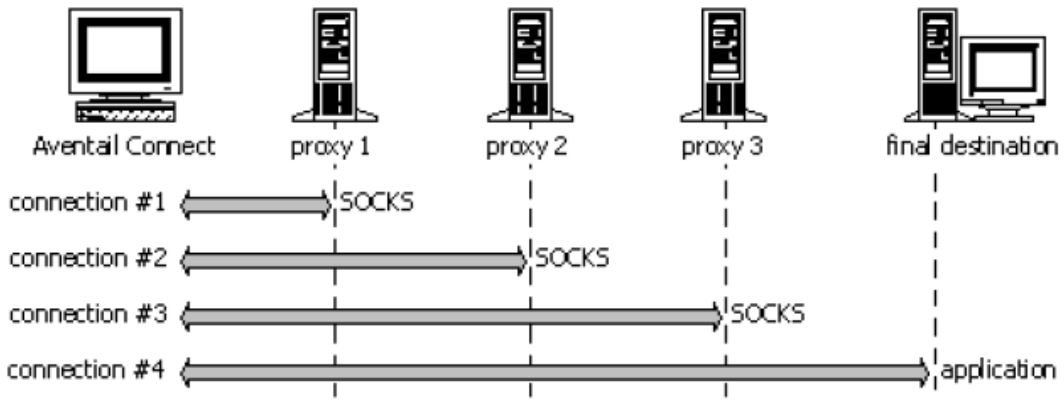
> The Aventail MultiProxy feature allows Aventail Connect to traverse multiple firewalls by making connections through successive proxy servers. Aventail Connect makes a connection with each proxy server individually. Each proxy server forms a link in a chain that connects Aventail Connect to the final 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.

Ex. 1009 (ACAG) at 59.

259. Aventail Connect manages the routing of these communications, and handles authentication, encryption and access parameters to each of the intermediary proxy servers (such as SOCKS servers or HTTP proxy servers).

260. Aventail describes the steps that are taken to make a connection using MultiProxy:

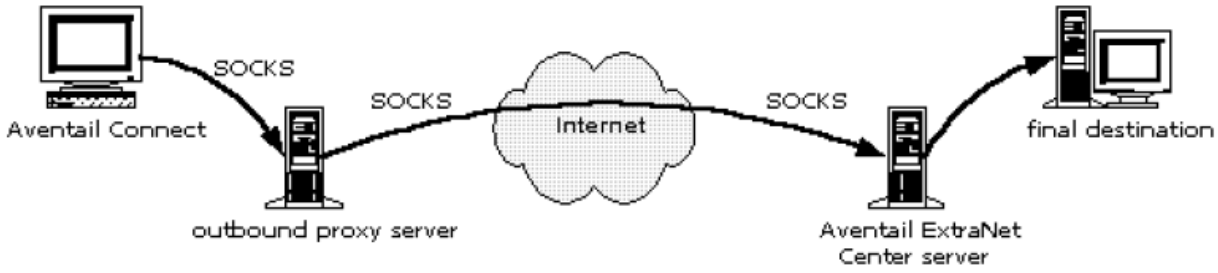
> The steps for making a connection using MultiProxy are:
>
> 1. The client application requests access to the destination server.
> 2. Aventail Connect establishes a connection with the outbound server (SOCKS server or HTTP proxy). Aventail Connect then sends the access request to the outbound server, 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.



Ex. 1009 (ACAG) at 60.

261. When a connection is made using Multiproxy, the proxy server computer (*i.e.*, the Aventail Extranet Server):

> 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.



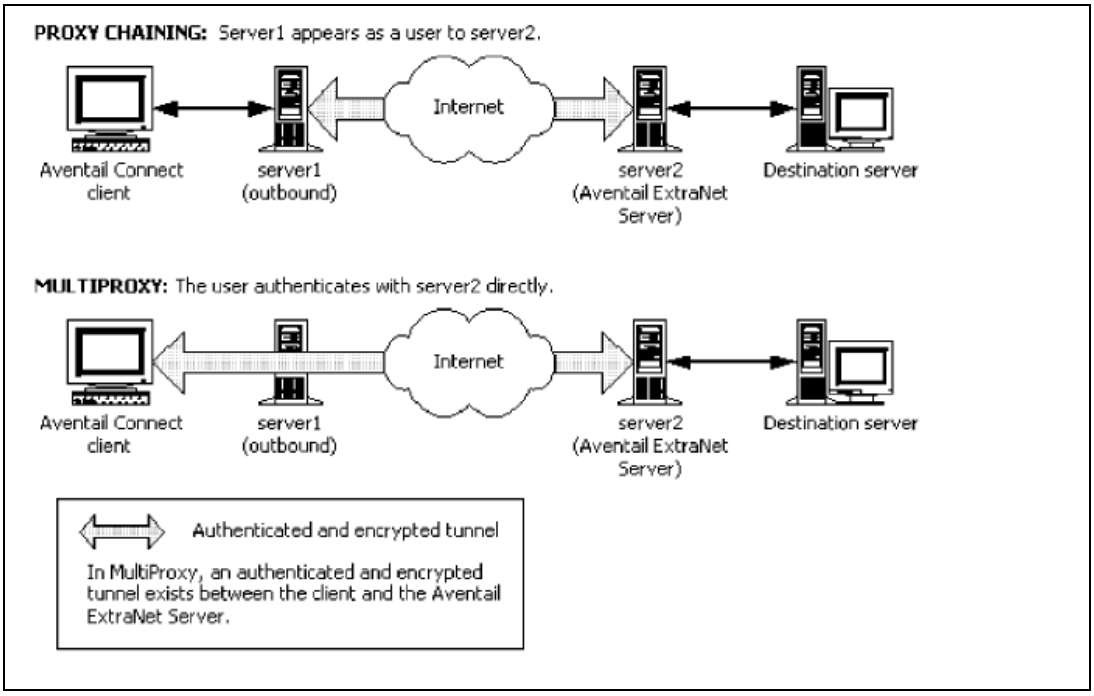
Ex. 1009 (ACAG) at 60.

262. Thus, Aventail Connect could route communications through one or more intermediary proxies. After defining the final destination and the extranet server locations, one or more intermediary proxy servers could be added (*i.e.*, a "destination") to the list of these proxies that Aventail Connect would use. All or some of the traffic from the computer running Aventail Connect could be routed through each destination, based on how Aventail Connect was configured. Ex. 1009 (ACAG) at 59-62.

263. In addition to Multiproxy, Aventail Connect also offered "proxy chaining" functionality. Here, Aventail Connect sends traffic to a specified intermediary proxy server, which forwards that traffic on to one (or more)

additional proxy servers. Ex. 1009 (ACAG) at 63-64. Authentication with the first proxy server would be handled by Aventail Connect, while each subsequent authentication would occur between the intermediary server and the next proxy server in the chain. Ex. 1009 (ACAG) at 63.

264. Aventail provides the figure shown below comparing the authentication and access rule variables for "MultiProxy" and "Proxy Chaining" routing:



Ex. 1009 (ACAG) at 63.

265. As can be seen, in the case of MultiProxy, Aventail Connect communicates "directly" with the ExtraNet Server even though there is an intermediary proxy gateway (server1) between the client and the Extranet. As the illustration shows, the connection does not start or stop at the intermediary proxy

server, showing that these proxies are capable of forwarding encrypted VPN traffic without impeding direct communication.

10. Secure Extranet Explorer

266. Another feature provided by Aventail Connect was the "Secure Extranet Explorer" (SEE), which allowed remote users running Aventail Connect to dynamically browse and access resources on the private network once a secure connection had been established. SEE is described as follows in Aventail:

> 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.

Ex. 1009 (ACAG) at 90.

267. Thus, a remote user who had successfully established a VPN to a private network using Aventail Connect would be able to see and access all of the network resources that user was authorized to access, just as if the remote user had a local connection to the network. *Id*.

268. Aventail Connect implemented SEE through a Windows Explorer shell extension called "Extranet Neighborhood." This shell extension allowed a Windows OS client to visually navigate resources (computers, servers, files, etc.)

on a private network to which the client has established a VPN connection. *See* Ex. 1009 (ACAG) at 90-101.

269. Aventail describes the Extranet Neighborhood shell extension as follows:

> 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.

Ex. 1009 (ACAG) at 90.

270. According to Aventail, the Extranet Neighborhood functionality was implemented by "redirecting" Windows communications in NetBIOS (NBT) to WinSock. Ex. 1009 (ACAG) at 91. Aventail Connect was thus able to handle these types of communications in the same manner that it handled DNS requests and TCP/IP traffic – by slipping itself in as an intermediary layer and intercepting requests. Ex. 1009 (ACAG) at at 91 ("To deliver a secured version of standard Windows browsing, Aventail Connect redirects NBT calls to WinSock.").

271. Using Extranet Neighborhood, a user running Aventail Connect could view and access a "dynamic list of available Windows hosts" (the secure computers on the private network). Ex. 1009 (ACAG) at 91. Aventail explains that in order to see this dynamic list, Aventail Connect had to be configured to retrieve information used by Windows to map network resources. Ex. 1009 (ACAG) at 91 ("To use Extranet Neighborhood in the browsing mode, you must configure Extranet Neighborhood to use WINS, and you must identify the IP address (hostname) 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.")

272. A user running Aventail Connect and using the Extranet Neighborhood feature would be able to see and access the same resources that

other users on the local or private network would see (subject, of course, to any user specific access restrictions). Aventail explains:

> 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.

Ex. 1009 (ACAG) at 91.

11. Domains Names of Hosts on the Private Networks Protected by an Aventail Extranet Server

273. I note that Aventail shows that private networks can have their own DNS servers that are different from the public DNS servers. S*ee* Ex. 1009 (ACAG) at 72-74. DNS servers on a private network protected by an Aventail Extranet server would be accessible only by computers on the private network or by computers that have created a VPN connection with the private network with the necessary authentication and encryption. *See* Ex. 1009 (ACAG) at 72-74. Thus the domain names serviced by the private DNS servers (e.g., those of hosts on the private network) would be accessible only to those external client computers

with a VPN connection—if a public DNS was asked to resolve those private domain names, an error indicating that the domain name could not be resolved would be returned.

B. U.S. Patent No. 6,496,867 to Beser

1. Overview of Beser

274. Beser describes methods and systems for using private IP addresses to establish a "virtual" tunneling association between two end devices, an originating device and a terminating device, over a public network with the aid of two network devices serving as routers or gateways (referred to as first network device and second network device) and a third network device providing a directory service (referred as a trusted-third-party network device). Ex. 1007 (Beser) at 7:62-8:-20. End devices include Voice over IP (VoIP) devices, media player devices, and personal computers, amongst others. Ex. 1007 (Beser) at 4:44-52.

275. In the Beser methods and systems, each end device is connected to a network device (*e.g.*, an edge router) via a private network or other network. Ex. 1007 (Beser) at 4:19-29. Such a network device allows the end device to establish a tunnel to another end device over a public network such as the Internet, via an intermediate second network device. Ex. 1007 (Beser) at Fig. 1, 4:19-42, 7:67-8:1.

276. Each end device is associated with a "unique identifier" that can be used to initiate a secure tunnel with other end devices. Ex. 1007 (Beser) at 10:37-

41. Beser explains that the unique identifiers can be domain names, E.164 numbers, email addresses, or a number of other identifiers. Ex. 1007 (Beser) at 10:55-11:8.

277. The trusted-third-party network device can be a modified DNS server that maintains a database associating each registered unique identifier with the IP address of its network device. Ex. 1007 (Beser) at 11:32-36, 11:45-58.

278. When an originating end device tries to connect to a terminating end device, it sends out a request to initiate a tunneling association that contains the unique identifier of the terminating end device. Ex. 1007 (Beser) at 9:64-10:6; *see id.* at 7:61-8:20. The request is received by a first network device, witch in turn routes the request to a trusted-third-party network device. Ex. 1007 (Beser) at 7:64-66, 8:21-22, 8:48-49, 10:22-23, 11:9-10. The trusted-third-party network device then identifies the network device associated with the terminating end device (*e.g.*, by looking up the identifier of the terminating end device in a database). Ex. 1007 (Beser) at 9:26-30, 11:30-58. Finally the trusted-third-party network device interacts with the first network device (associated with the originating end device) and the second network device (associated with the terminating end device) to negotiate a private IP address to assign to each end device. Ex. 1007 (Beser) at 9:26-62, 11:59-63. Private IP addresses are addresses

"reserved for use in private networks that are isolated from a public network such as the Internet" and "are not globally routable." Ex. 1007 (Beser) at 11:62-65.

279. The first and second network devices will use the private IP addresses to route packets to the end devices connected to them. Ex. 1007 (Beser) at 3:4-9, 22:4-22. Beser explains that the negotiation of the private IP addresses is conducted with messages exchanges between the first and second network devices and the trusted-third-party device. Ex. 1007 (Beser) at 12:10-13, 12:55-58, Fig. 9. The IP packets exchanged during the negotiation have as source or destination only the IP addresses of the first, second, or trusted-third-party network device. Ex. 1007 (Beser) at 12:10-13. Indeed, referring to the VoIP embodiment, Beser notes "neither the private nor any public IP 58 addresses for the ends of the VoIP association appear in the source 88 or destination 90 address fields of the IP 58 packets that comprise the negotiation." Ex. 1007 (Beser) at 12:6-9. Thus, someone who observes the source and destination fields of such packets will not learn the identity of the endpoint devices, which appears only in payloads of such packets. *Id.* For additional security, the payloads can be encrypted so the identities of the endpoint devices remain hidden also from someone who eavesdrops the payload of such packets. Ex. 1007 (Beser) at 11:22-25, 18:2-5, 20:11-14.

280. The tunneling associations described in Beser provide security by hiding (obfuscating) the terminating addresses of the sending and receiving

computers or devices. *See, e.g.*, Ex. 1007 (Beser), abstract. This makes the tunneling associations "private" or "anonymous" because they cannot be discovered on the public network. Ex. 1007 (Beser) at 3:4-9, 8:15-20, 12:6-19.

281. IP tunneling techniques were well known in February 2000, and could used to create "virtual tunnels" between nodes. For example, Beser explains:

> One service that can be performed in the application layer is that of initiating and maintaining a virtual tunnel. Virtual tunneling techniques are known to those skilled in the art. Examples of virtual tunneling include Internet Encapsulation Protocol tunneling, Generic Route Encapsulation ("GRE"), and IP in IP tunneling. All three tunneling techniques are based on encapsulating a data packet of one protocol inside a data packet of another protocol. For more

Ex. 1007 (Beser) at 6:58-65.

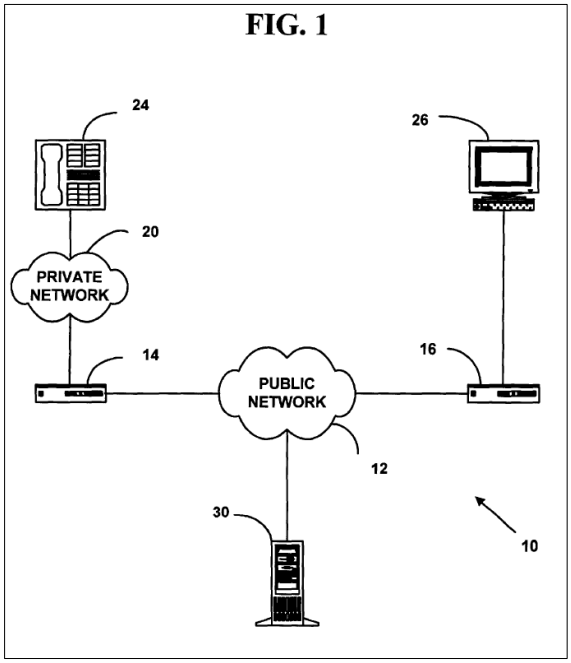
282. The methods and systems described in Beser are designed to work with conventional processes and techniques. *See, e.g.*, Ex. 1007 (Beser) at 4:55-5:2; *id.* at 1:54-56, 2:18-27.

283. Beser explains its system can be integrated into any standards-compliant pre-existing (legacy) equipment and systems. Ex. 1007 (Beser) at 4:44-5:2. For example, the Beser system can be integrated into pre-existing corporate networks. Ex. 1007 (Beser) at 10:57-66 ("the user of terminating telephony device 26 may have moved from one office to another office while still retaining the same electronic mail address"). In such an implementation, the devices are not limited to communicating using the Beser IP tunnel technique. Each end device could

communicate with conventional devices (*e.g.*, a standard website) without creating a tunnel. The network devices, such as an edge router or gateway, would handle both conventional IP traffic as well as Beser tunneling associations.

2. Basic Components

284. One method Beser describes establishes an IP tunneling association between an originating (24) and terminating (26) end devices, with the aid of a first network device (14), a second network device (16), and a trusted-third-party network device (30). Ex. 1007 (Beser) at 2:46-67, 7:65-8:12 This scenario is illustrated in Fig. 1:



Ex. 1007 (Beser) at Fig. 1.

285. The Beser system is compatible with many types of devices and data, Beser describes many examples of those devices and data. I describe those below.

d) Originating and Terminating End Devices

286. Beser explains that the originating and terminating end devices (24 and 26) can be "telephony devices" or "multimedia devices," and gives several examples of each. Ex. 1007 (Beser) at 4:43-47.

287. As examples of telephony devices, Beser identifies "VoIP devices (portable or stationary)." Ex. 1007 (Beser) at 4:50-52. A person of ordinary skill in the art would have understood that these devices include VoIP-enabled telephones, mobile phones, and PDAs.

288. As another example of telephony devices, Beser identifies "personal computers running facsimile or audio applications." Ex. 1007 (Beser) at 4:51-52. A person of ordinary skill in the art would have understood that a "personal computer" would include both stationary desktop computers and portable laptop computers.

289. As examples of multimedia devices, Beser identifies "Web-TV sets and decoders, interactive video-game players, or personal computers running multimedia applications." Ex. 1007 (Beser) at 4:47-49. Again, a person of ordinary skill would have understood a "personal computer" to include both desktop and portable laptop computers. That person also would have understood "multimedia applications" to include web browsers, audio and video players, photo browsers, and audio- and video-conferencing software. Web-TV devices and

multimedia applications could be used to connect to and downloading from a server.

290. Beser explains that end devices can be "portable" devices (Ex. 1007 (Beser) at 4:50-52), and that data may be sent to and from an end device (*e.g.*, a portable VoIP device) using data transmission standards that were developed by the "Wireless Application Protocol ('WAP') Forum." Ex. 1007 (Beser) at 4:55-62. It was well-known that the WAP standards define protocols for transmitting data packets over cellular networks. *See* Ex. 1038 (W3C WAP) at 2-3. A person of ordinary skill in the art considering these portions of Beser would have understood that the end devices in the Beser system could be mobile phones or other mobile devices that communicate over a cellular network.

291. Indeed, the WAP Forum was well known in February 2000:

> The WAP Forum is an industry association that has developed the de-facto world standard for wireless information and telephony services on digital mobile phones and other wireless terminals. Handset manufacturers representing 95 percent of the world market across all technologies have committed to shipping WAP-enabled devices.

Ex. 1038 (W3C WAP) at 2-3.

292. It was well known in February 2000 that WAP "specifies an application framework and network protocols for wireless devices such as mobile telephones, pagers, and personal digital assistants (PDAs). The specifications extend and leverage mobile networking technologies (such as digital data

networking standards) and Internet technologies (such as XML, URLs, scripting, and various content formats)." Ex. 1039 (WAP Architecture Spec) at 3.

293. The network devices and components described in Beser include computers and other programmable devices that have processing elements and storage media containing code that configures how those devices function. Ex. 1007 (Beser) at 4:19-54, 5:15-47. It was known in February 2000 that personal computers include processors and storage devices. Telephony devices and multimedia devices such as those described in Beser include memory and processing elements.

e) First and Second Network Devices

294. Beser explains that the first and second network devices (14 and 16) can be "edge routers," cable modems, "modified routers" or "modified gateways." Ex. 1007 (Beser) at 4:7-8, 4:19-42. A person of ordinary skill in the art would have understood that these devices are standard networking devices (routers, modems, etc.) modified to implement the methods disclosed in Beser.

295. A router is an intermediary computer that directs or "routes" IP traffic across the network. For example, Beser explains that edge routers "route[] data packets between one or more networks such as a backbone network (*e.g.*, a public network 12) and Local Area Networks (*e.g.*, private network 20)." Ex. 1007 (Beser) at 4:21-24; *see also* ¶116 above.

296. Typically, a router operates at the network layer of the protocol stack. *See* Ex. 1007 (Beser) at Fig. 2, 6:7-10 ("Above both the data-link layer is an Internet Protocol ('IP') layer 58. The IP layer 58, hereinafter IP 58, roughly corresponds to OSI layer 3, the network layer, but is typically not defined as part of the OSI model."). Because a router typically receives all IP packets on the data link layer, it receives all IP packets passed through it, even those not addressed to the router.

297. The term "modem" means "modulator-demodulator." *See* ¶80. It was well known in February 2000 that cable modems would transmit data by modulating and demodulating a signal over a coaxial cable line. Ex. 1019 (MS Dictionary) at 69-70 (defining "cable modem"). It was also well known that cable modems utilized frequency division multiplexing ("FDM") and time division multiplexing ("TDM") to enable the sharing of a line by multiple customers. Ex. 1048 (DOCSIS) at 21.

298. The first and second network devices are also called "gateway" devices in Beser because they are connected to both a private network and the public Internet. *See* Ex. 1007 (Beser) at 4:7-8, Fig. 1.

299. Beser shows the first and second network devices run a "tunneling application" or are otherwise modified to support initiating an IP tunnel. Ex. 1007 (Beser) at 8:39-43. As part of the Beser process, each network device monitors the

data traffic passing through it to determine if any data packets correspond to a request to initiate a tunneling association. Ex. 1007 (Beser) at 8:21-29. This can be done by monitoring for an "indicator" that the datagram is associated with a higher layer such as "a distinctive sequence of bits at the beginning of the datagram." Ex. 1007 (Beser) at 8:34-39. The distinctive sequence of bits "indicates to the tunneling application that it should examine the request message for its content and not ignore the datagram." Ex. 1007 (Beser) at 8:39-43.

300. If the network device determines that a particular data packet contains an indicator that it is associated with the tunneling application, the device will send the packet to the trusted-third-party network device for special processing. Ex. 1007 (Beser) at 8:29-43. Otherwise, the tunneling application will "ignore the datagram," meaning the network device will process the data packet normally. Ex. 1007 (Beser) at 8:39-44.

301. Beser also shows the tunneling application allows the first and second network devices to communicate with the trusted-third-party network device and to negotiate private IP addresses for the end devices. Ex. 1007 (Beser) at 11:26-30.

302. Beser shows that each first and second network device is modified to maintain a set of special routing tables for handling IP tunnel traffic. Ex. 1007 (Beser) at 12:28-36, 21:63-22:22; *see id.* at 23:50-25:34.

f) Trusted-Third-Party Network Device

303. Beser explains that the trusted-third-party network device (30) can be "a back-end service, a domain name server, or the owner/manager of database or directory services and may be distributed over several physical locations." Ex. 1007 (Beser) at 11:32-36; *see also id.* at 4:9-11. Each of these servers and services was conventional and well-known in February 2000.

304. It was well-known in February 2000 that both "domain name servers" (DNS servers) and "directory services" are computers or services that track the location of resources or computers on a network, and they are configured to respond to look-up requests from client devices by providing address data for resources or computers. *See, e.g.*, Ex. 1019 (Microsoft Computer Dictionary 4th ed.) at 142, 148-49.

305. It was well-known in February 2000 that that a "back-end" service is a service that interfaces with an application and that is not directly used by a user. *Id.* at 39. DNS servers and directory services typically are back-end services because users usually do not directly interact with those servers. Instead users interact with application programs, and DNS servers and directory services provide information to the application programs.

306. The functionality of a DNS server was extremely well-known by February 2000. The primary function of a DNS server was correlate IP addresses

with domain names, and to respond to look-up requests by returning the appropriate address information for a requested name. *See* ¶126 above; *see also* Ex. 1001 ('705 patent) at 39:1-3 ("Conventional Domain Name Servers (DNSs) provide a look-up function that returns the IP of a requested computer or host."). If the IP address is unknown, a DNS server would not resolve the address and instead return an error message.

307. Beser describes the trusted-third-party network device as a conventional device that is modified to include a tunneling application or otherwise support creating IP tunnels. Ex. 1007 (Beser) at 8:65-9:1, 11:45-58. So, if the trusted-third-party network device were a "domain name server" (Ex. 1007 (Beser) at 11:32-36), it would be a conventional domain name server modified to include additional Beser functionality.

308. Beser explains that the trusted-third-party network devices stores and maintains a database that maps each registered unique identifier to one or more associated IP addresses. Ex. 1007 (Beser) at 11:45-58. Beser describes an example where the trusted-third-party network device includes a directory service for subscribers, which uses a database to associate each registered E.164 number (unique identifier) with the IP address of the corresponding second network device and the IP address of the end device. Ex. 1007 (Beser) at 11:45-58. Beser also explains that this database can be used to track other types of unique identifiers,

including domain names used as unique identifiers. Ex. 1007 (Beser) at 11:55-58. In other words, Beser explains that instead of the E.164 number of the terminating device being associated with the IP address of the corresponding second network device, a domain name could be used for the unique identifier of the terminating device. Ex. 1007 (Beser) at 10:55-11:5.

309. Like the first and second network devices, the trusted-third-party network device runs a tunneling application that can detect packets that need special processing. Ex. 1007 (Beser) at 8:57-9:1 ("the distinctive sequence of bits indicates to the tunneling application that it should examine the informing message for its content and not ignore the datagram"). If a data packet needs special processing, it would be handled by the tunneling application. Ex. 1007 (Beser) at 8:65-9:1. Otherwise the tunneling application would "ignore the datagram," meaning the trusted-third-party network device would process message normally. Ex. 1007 (Beser) at 9:1.

310. When the trusted-third-party network device receives a request to initiate a tunneling association, it uses the unique identifier in the request to look-up the corresponding IP address in its database of registered unique identifiers. Ex. 1007 (Beser) at 11:26-36, 11:45-55. To initiate the secure IP tunnel, the trusted-third-party network device will look-up the IP address of the corresponding second network device. Ex. 1007 (Beser) at 9:6-8, 11:26-36.

311. The trusted-third-party network device is configured to negotiate with the first and second network devices to establish private IP addresses for the end devices. Ex. 1007 (Beser) at 9:26-37, 12:38-54.

3. IP Tunnel

312. The Beser systems can be configured to create IP tunnels for transmitting many different types of data. For example, Beser describes transmitting VoIP traffic, "multimedia" content (*e.g.*, video or audio), or content for web pages (*e.g.*, delivered to WebTV devices or decoders or personal computers). Ex. 1007 (Beser) at 4:47-52. The data can be formatted according to many different protocols, such as HTTP (for web data), H.323, and FTP. Ex. 1007 (Beser) at 7:10-15 ("The payload field 84 of the IP 58 packet 80 typically comprises the data that is sent from one network device to another. However, the payload field 84 may also comprise network management messages, such as ICMP 56 messages, or data packets of another protocol such as UDP 60, SNMP 62, TFTP 64, or DHCP 66."); *id.* at 6:24-57; *id.* 9:64-10:2 (H.323).

313. Beser explains its systems are compatible with the H.323 standards. Ex. 1007 (Beser) at 9:64-10:2. A person of ordinary skill in the art would have understood that multimedia communications compliant with the H.323 standard included video conference communications. Ex. 1040 (H.323) at 11 ("H.323

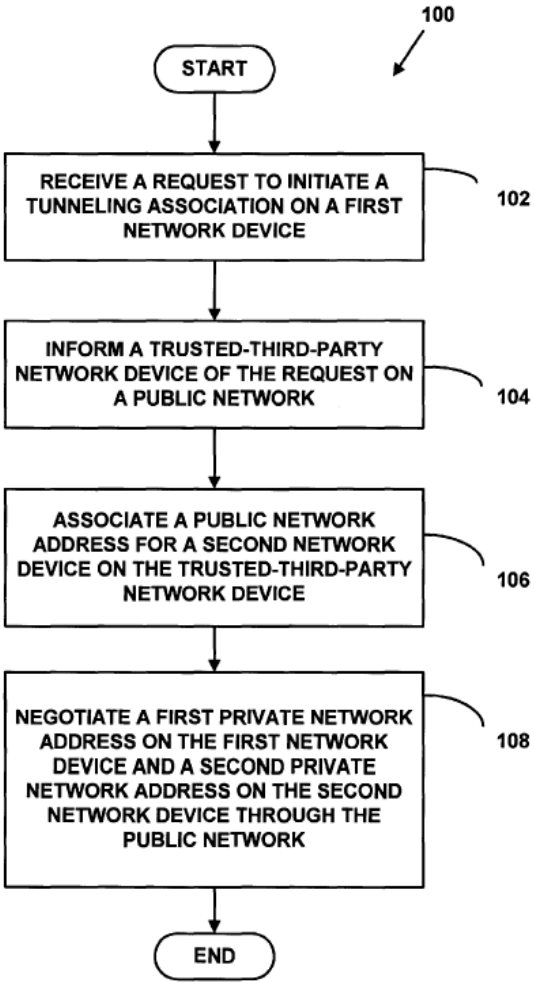
terminals provide audio and optionally video and data communications capability in point-to-point or multipoint conferences").

314. The Beser scheme may operate at different protocol levels. For example, Beser explains that the request to initiate an IP tunnel can be received in a higher layer of a protocol stack for the first network device such as the transport layer or the application layer. Ex. 1007 (Beser) at 8:22-29.

4. Establishing an IP Tunnel

315. The Beser tunneling method is illustrated in general terms in Fig. 4, shown below:

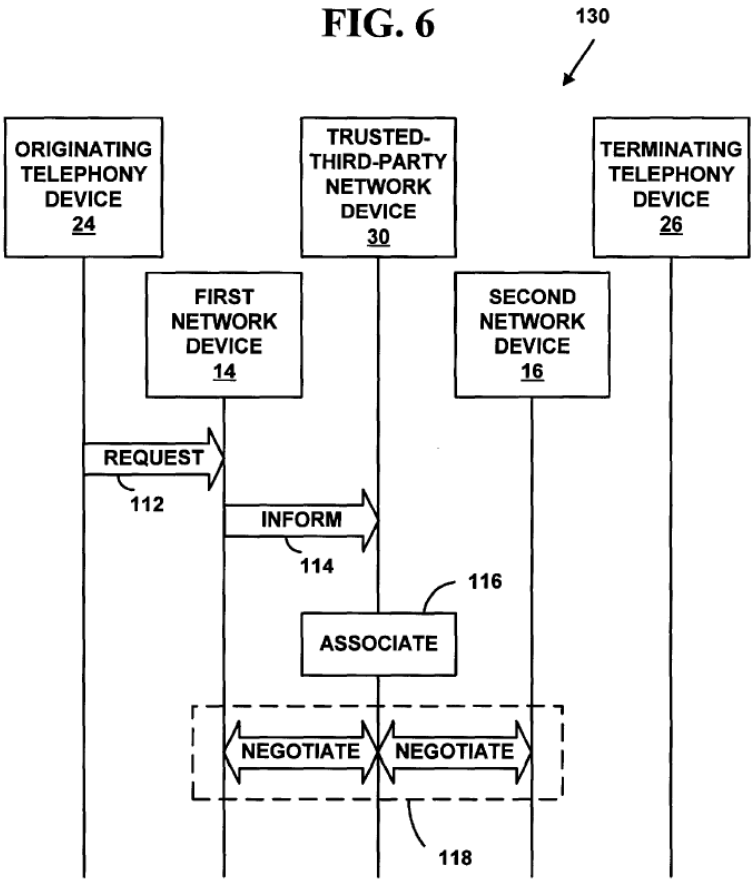
FIG. 4



Ex. 1007 (Beser) at Fig. 4.

g) Request Containing A Unique Identifier

316. The Beser process starts with "receiving a request to initiate the tunneling connection on a first network device at step 102." Ex. 1007 (Beser) at 7:63-66. An originating end device, or an application running on the device, will generate the request. Ex. 1007 (Beser) at 9:64-10:41. This is generally illustrated in Fig. 6 (*see* step 112):



Ex. 1007 (Beser) at Fig. 6.

317. In Beser, the first network device intercepts and evaluates all of the data packets passed through it. If a packet contains a request to initiate an IP tunnel, it will contain an indicator, such as a distinctive sequence of bits, to inform the first network device that the packet needs special processing. Ex. 1007 (Beser) at 8:33-43. I described this indicator in more detail above. *See* ¶¶ 299-300, *above*. If the first network device determines a data packet contains a request to initiate an IP tunnel, it sends the request to a trusted-third-party device (*see* step 114 in Fig. 6). Ex. 1007 (Beser) at 8:21-50. Otherwise it processes the packet normally. *Id.*

318. Beser explains that the request from the originating end device will include a "unique identifier" for the terminating end of the tunneling association. Ex. 1007 (Beser) at 8:1-3, 10:37-11:8. This unique identifier specifies the destination or "target" tunneling association, not the trusted-third-party network device. Ex. 1007 (Beser) at 10:37-38.

319. The unique identifier can take many forms, such as "a dial-up number, an electronic mail address, or a domain name." Ex. 1007 (Beser) at 10:40-41; *see also* 10:66-11:2 ("There are many other possibilities for the unique identifier, e.g. employee number, social security number, driver's license number, or even a previously assigned public IP 58 address."). A unique identifier could be a domain name (*e.g.* if a web browser made a request for a website) or a dial-up number (*e.g.*, if a VoIP application made a request to connect to a telephone number), and the Beser system would function in the same way. Ex. 1007 (Beser) at 10:37-42, 10:55-57.

320. As an example of the Beser system, a VoIP application running on a portable telephony device could initiate an IP tunnel by sending out a request that included the domain name associated with the terminating end device. Ex. 1007 (Beser) at 4:50-52, 10:55-66. As another example, the request could be sent by a personal computer running a multimedia application or a web browser, and it could

include the domain name of the web server containing multimedia content. Ex. 1007 (Beser) at 4:47-54, Fig. 1.

321. Beser also explains that the transmission of a request containing a unique identifier "may require encryption or authentication to ensure that the unique identifier cannot be read on the public network." Ex. 1007 (Beser) at 11:22-25; *see also* 18:2-5, 20:11-14.

322. If a data packet is a request to initiate a tunnel, the first network device sends the request to the trusted-third-party network device. Ex. 1007 (Beser) at 8:29-49. The trusted-third-party network device evaluates the request (including the unique identifier in the request) and takes action based on that evaluation. *See, e.g.*, Ex. 1007 (Beser) at 11:45-58, 17:45-49.

323. One way in which the trusted-third-party network device may evaluate the request is to compare the unique identifier to a database of entries (*e.g.*, a database of subscribers) to determine whether the unique identifier correlates to an IP address associated with a second network device or a terminating end device, *i.e.*, a user or end device that is able to communicate using an IP tunnel. *See, e.g.*, Ex. 1007 (Beser) at 9:6-11, 11:26-36, 11:45-58, 17:45-49. The database contains the list of authorized users or devices, and each entry in the database may include the unique identifier, the "IP 58 address of a particular

second network device 16," and the "public IP 58 addresses for the terminating telephony device 26." Ex. 1007 (Beser) at 11:49-52.

324. If the unique identifier in the request specifies a secure destination (*e.g.*, the unique identifier corresponds to a second network device or terminating end device), the trusted-third-party network device will automatically establish a tunneling association by negotiating with the first and second network devices. *See, e.g.*, Ex. 1007 (Beser) at Fig. 4, 11:9-44, 11:58-12:19.

325. In this process, the trusted-third-party network device determines whether the unique identifier corresponds to a secure destination by reference to an internal database. Ex. 1007 (Beser) at 11:32-36, 11:48-52. The '705 and '009 specifications describe an analogous process, where a DNS proxy server determines whether access to a secure site has been requested by reference to an internal table of such sites. Ex. 1001 ('705 Patent) at 40:3-7; Ex. 1003 ('009 Patent) 40:41-15.

326. Beser does not explicitly describe what would happen if the unique identifier was not listed in the trusted-third-party network device's database of registered identifiers. Configuring the trusted-third-party network device to handle such an error condition would have been a simple engineering design choice, which would have been routinely made by a person of ordinary skill in the art. The tunneling application on the trusted-third-party network device could (i) do

nothing, (ii) return an error code, or (iii) pass the unique identifier to a conventional domain name server or directory service for resolution. In any scenario, without a valid unique identifier, the trusted-third-party network device could not negotiate an IP tunnel.

327. As explained above, the trusted-third-party network device can be a DNS server. *See* ¶303 above. In that case, the database of registered unique identifiers could be maintained separate from the device's DNS tables or it could be incorporated into the DNS tables. Ex. 1007 (Beser) at 11:52-55 ("Many data structures that are known to those skilled in the art are possible for the association of the unique identifiers and IP 58 addresses for the second network devices 16.").

328. Where the trusted-third-party network device incorporated the registered unique identifiers into its DNS tables, if the unique identifier did not correspond to a registered end device, the trusted-third-party network device would resolve the domain name and return the public IP address of the second network device, as this is what DNS servers do. *See, e.g.*, Ex. 1001 ('705 Patent) at 39:1-3 ("Conventional Domain Name Servers (DNSs) provide a look-up function that returns the IP address of a requested computer or host."); *see also* ¶¶122-128 above.

329. Where the trusted-third-party network device maintained a separate database, if the unique identifier was not listed in the database of registered

identifiers, it would have been obvious to configure the trusted-third-party network device to pass the request to the conventional DNS module to resolve the domain name and return the public IP address of the second network device.

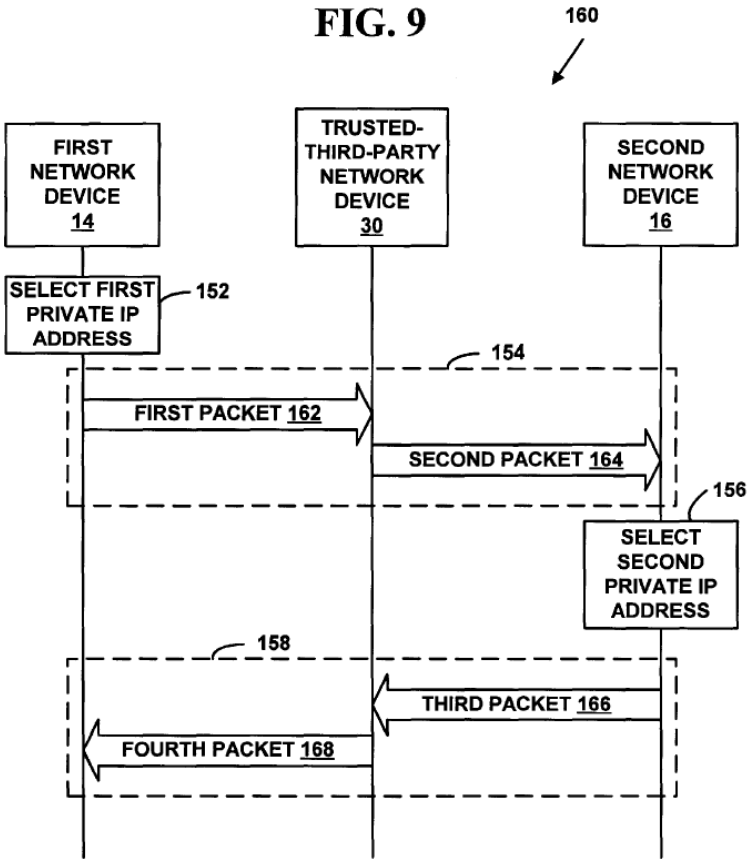
330. If the unique identifier in the request specifies a secure destination, the trusted-third-party network device will negotiate private IP addresses for the end devices. Ex. 1007 (Beser) at 9:26-30, 11:59-62, 12:38-54. Although Beser explains that, in some embodiments, the trusted-third-party network device does not need to be involved in the negotiation, it describes a preferred embodiment where it is. Ex. 1007 (Beser) at 9:29-35. I describe this process in more detail in the next section.

331. Where the trusted-third-party network device negotiates the private IP addresses, it will transmit the unique identifier to the second network device. Ex. 1007 (Beser) at 11:9-22. The trusted-third-party network device will also return to the first network device a public IP address associated with the unique identifier. Ex. 1007 (Beser) at 14:51-57, 17:50-59.

332. The unique identifier may be used as an identifier for the tunneling association in subsequent communications between the second and first network devices. Ex. 1007 (Beser) at 14:7-13.

h) Negotiation of Private IP Addresses

333. In the Beser scheme, the trusted-third-party network device negotiates private IP addresses for the originating and terminating end devices by communicating with the first and second network devices over a public network. Ex. 1007 (Beser) at 9:26-30, 11:59-63. This is generally illustrated in Fig. 9:



Ex. 1007 (Beser) at Fig. 9.

334. Beser's scheme employs standard IP packets to establish an IP tunnel. Ex. 1007 (Beser) at 7:7-26. Standard IP packets have a "header" and a "payload." Ex. 1007 (Beser) at 7:9-10. The header includes a "source" and a "destination"

address. Ex. 1007 (Beser) at 7:15-17. The payload typically includes the data that is carried by the IP packet. Ex. 1007 (Beser) at 7:10-12.

335. Each of the packets in this negotiation is sent to or from the trusted-third-party network device, and each contains only public addresses for the first, second, and trusted-third-party network devices in the "source" and "destination" address fields. Ex. 1007 (Beser) at 13:13-18, 13:33-38; 14:2-7, 14:19-23. This process ensures that the true IP addresses of the end devices are not revealed. Ex. 1007 (Beser) at 12:6-9.

336. The first network device selects a "first private IP address," which it assigns to the originating end device. Ex. 1007 (Beser) at 12:63-13:9. This "first private IP address" is transmitted to the trusted-third-party network device in the "first" packet of Figure 9. Ex. 1007 (Beser) at 13:13-25.

337. The trusted-third-party network device then transmits a "second" packet, which it sends to the public IP address of the second network device. Ex. 1007 (Beser) at 13:38. The packet contains the private IP address for the originating device and the public IP address for the first network device. Ex. 1007 (Beser) at 13:38-42.

338. Next, the second network device selects a "second private IP address," which it assigns to the terminating end device. Ex. 1007 (Beser) at 13:49-63. The second network device transmits a "third" packet to the trusted-third-party network

device. Ex. 1007 (Beser) at 14:2-14. The third packet contains the private IP address for the terminating end device as well as "an identifier for the tunneling association," such as the unique identifier. Ex. 1007 (Beser) at 14:7-12.

339. The trusted-third-party network device then transmits a "fourth" packet, which it sends to the public IP address of the first network device. Ex. 1007 (Beser) at 14:19-23. The packet contains the private IP address for the terminating end device and the public IP address for the second network device. Ex. 1007 (Beser) at 14:23-27.

340. The first network device then transmits the private IP addresses to the originating end device. Ex. 1007 (Beser) at 14:57-62, 21:48-52. The second network device transmits the private addresses to the terminating end device. *Id.* As a result of Beser's scheme, the originating and terminating end devices obtain both private IP addresses:

> The assignment of private network addresses to the ends of the tunneling association may also include transmitting the private network addresses to the network devices at the ends of the tunneling association where the private network addresses are stored on these end devices. For example, the originating network device **24** may store the private network addresses for the originating and terminating ends of the tunneling association on the originating network device **24**. In this manner, any packet sent from the originating network device **24** to the first network device **14** may include the private network addresses of the originating and terminating ends of the tunneling association. Similarly, any packet sent from the terminating network device **26** to the second network device **16** may include the private network addresses of the originating and terminating ends of the tunneling association.

Ex. 1007 (Beser) at 21:48-62.

341. Receipt of the two private IP addresses would be an indication to the originating end device that the trusted-third-party network device has established an IP tunnel. Ex. 1007 (Beser) at 21:48-62.

342. After an IP tunnel is established according to Beser's process, the originating and terminating end devices may communicate securely by using their private IP addresses. Ex. 1007 (Beser) at 21:52-62. The originating end device will use the private IP addresses to securely transmit packets to the terminating end device. Ex. 1007 (Beser) at 21:52-62, 22:2-22.

343. Beser explains that "any packet sent from the originating network device 24 to the first network device 14 may include the private network addresses of the originating and terminating ends of the tunneling association." Ex. 1007

(Beser) at 21:56-59. What this means is that the IP packet will have the originating end device's private IP address in the source IP address field and the terminating end device's private IP address in the destination IP address field. Both private IP addresses are used by the originating end device to transmit data to the terminating end device. *See* Ex. 1007 (Beser) at 7:7-26, 21:56-62.

344. To maintain anonymity, the first and second network devices modify the headers on packets sent between the originating and terminating end devices to replace private IP addresses with public addresses when transmitting data over the public network. Ex. 1007 (Beser) at 22:4-22. They reverse the process when receiving data packets from the public network before transmitting them across the local network. *Id.*

345. The creation of an IP tunnel using Beser's technique is transparent to the user, who simply initiates the request by taking an action (*e.g.*, entering the website destination of a WebTV source or initiating a VoIP call). Ex. 1007 (Beser) at 4:43-54; 10:22-36.

C. RFC 2401, the IPsec Protocol

1. Overview of RFC 2401

346. RFC 2401 describes IPsec, a standard that enables a network device to securely communicate with another network device over a network supporting the IP protocol (see the summary of IPsec in Section 3 of RFC 2401). In particular,

IPsec supports the encrypted communication between a host and a security gateway device at the boundary of a public network and a private network (Section 3.1 of RFC 2401). A person of skill in the art would have been familiar with RFC 2401, the techniques that it proposed, and the problems that its techniques were designed to solve.

347. IPsec extends the IP protocol to provide confidentiality (assurance that only authorized parties can access the transmitted data), sender authentication (assurance of the identity of the party transmitting the data), and data integrity (assurance that the transmitted data has not been altered while in transit). Confidentiality is achieved by means of encrypting the payload of an IP packet with a symmetric cryptosystem (see ¶¶130-131). Sender authentication and data integrity, jointly referred to as authentication, are achieved by adding to the IP packet a keyed cryptographic hash value. IPsec operates at the network layer. Thus, it is transparent to applications, which operate a higher layer. IPsec requires the sender and recipient to establish the shared keys used for encryption and for authentication and integrity. Ex. 1008 (RFC 2401) at 6, 51.

348. IPsec can be integrated into end devices, which RFC 2401 calls "hosts," and it also can be integrated into intermediary devices, which RFC 2401 calls "security gateways." Ex. 1008 (RFC 2401) at 6. IPsec can be used to protect a "path" between any combination of these devices – between a pair of hosts,

between a pair of security gateways (*e.g.*, intermediary devices such as routers), or between a security gateway and a host. *Id.* RFC 2401 explains that IPsec can be implemented into existing devices, such as via edge routers or gateway network devices. *Id.* at 7-8.

349. For each device, an administrator or user can specify a set of security rules, and that device will automatically apply a set of security service to packets that satisfy the rule. Ex. 1008 (RFC 2401) at 5-6. In each IPsec-complaint device, an IPsec module monitors all inbound and outbound IP traffic, and will automatically apply security services based on the predefined security rules (*e.g.*, automatically encrypt traffic to or from a certain IP address). *Id.* at 5-6, 14-15.

350. IPsec uses two protocols to provide security: Authentication Header (AH) and Encapsulating Security Payload (ESP). Ex. 1008 (RFC 2401) at 6. AH provides authentication (i.e., sender authentication and data integrity). *Id.* ESP provides data encryption and may also provide authentication (i.e., sender authentication and data integrity) among other services. *Id.* The Authentication Header protocol is described in detail in RFC 2402. The Encapsulating Security Payload protocol is described in detail in RFC 2406.

351. Each packet protected using IPsec includes an IPsec header (either an AH or ESP header) that is placed after an IP header but before the data payload. Ex. 1008 (RFC 2401) at 6, 9. The header describes how the packet is protected,

and contains the information necessary to authenticate the packet or provide other security services. Ex. 1008 (RFC 2401) at 6.

352. RFC 2401 explains that a unidirectional IPsec connection from a device to another device is called a "Security Association" (SA). Ex. 1008 (RFC 2401) at 8. Each SA is uniquely identified by a Security Parameter Index (SPI), IP Destination Address, and a security protocol (AH or ESP) identifier. *Id.* For a bidirectional IPsec connection between two devices, two SAs need to be established.

353. As RFC 2401 explains, the AH and ESP protocols supports two modes of use: "transport" mode and "tunnel" mode. Ex. 1008 (RFC 2401) at 7.

354. In "transport" mode, a host device will apply IPsec processing to data before adding the IP header. Ex. 1008 (RFC 2401) at 9. The IPsec header will be added at the front of the data payload, and then a normal IP header is added to the front. *Id.* A transport mode SA can only be utilized between two end devices. *Id.*

355. In "tunnel" mode, IPsec processing is applied to an entire IP packet. Ex. 1008 (RFC 2401) at 9. An IPsec header in added to the front of an IP packet and encapsulates the entire packet. *Id.* A new IP header is then added to the outside of the IPsec header. *Id.* This process creates an IP tunnel by encapsulating one IP packet inside of another packet. A tunnel mode SA can be used by both end devices and intermediate devices. *Id.*

356. IPsec can be configured to automatically encrypt certain IP traffic by setting a security rule. Ex. 1008 (RFC 2401) at 6, 14-15. An administrator can create a rule that IP traffic sent to a certain IP address or a certain range of IP addresses must be encrypted. *Id.* The administrator can control the granularity of protection by specifying the encryption algorithm, the key length, and other details. *Id.*

357. To provide end-to-end encryption between hosts using transport mode, a data stream will be split into segments. Each segment will be encrypted using a key known only to the sending host and receiving host and an ESP header will be added to the front of the encrypted segment. The sending host device will create an IP packet addressed to the receiving end host device that includes in the payload the ESP header and the encrypted segment. Ex. 1008 (RFC 2401) at 9-11.

358. RFC 2401 explains that an IPsec implementation will include mechanisms for creating and distributing cryptographic keys that will be used for encryption and authentication. Ex. 1008 (RFC 2401) at 6-7. RFC 2401 explains that several conventional techniques can be used to for this purpose, including the Internet Key Exchange (IKE) Protocol (RFC 2409) or a Key Distribution Center (KDC):

> Because these security services use shared secret values (cryptographic keys), IPsec relies on a separate set of mechanisms for putting these keys in place. (The keys are used for authentication/integrity and encryption services.) This document requires support for both manual and automatic distribution of keys. It specifies a specific public-key based approach (IKE -- [MSST97, Orm97, HC98]) for automatic key management, but other automated key distribution techniques MAY be used. For example, KDC-based systems such as Kerberos and other public-key systems such as SKIP could be employed.

Ex. 1008 (RFC 2401) at 7.

359. The IKE protocol was well-known to those of ordinary skill in the art, and it describes how two devices can mutually agree on cryptographic methods and keys to be used in a subsequent secure communication session that is encrypted and/or authenticated. *See* Ex. 1008 (RFC 2401) at 3-4, 7. I describe the IKE protocol in ¶¶139-140 above.

360. RFC 2401 also explains that several other conventional techniques can be used to distribute keys, including use of a "KDC" or Key Distribution Center. Ex. 1008 (RFC 2401) at 7. As I explained in ¶¶137-138 above, it was well-known that a KDC is a trusted entity or device that distributes keys and authentication information. A KDC typically is a centralized device or set of devices.

361. RFC 2401 explains that, in a security gateway implementation, each outbound IP packet is evaluated to determine what processing is required according to existing or created SAs, *i.e.*, authentication and/or encryption. Ex.

1008 (RFC 2401) at 30. If the packet requires IPsec processing, then authentication and encryption are applied automatically to create the IPsec relationship with the destination. *Id*. If the packet does not require IPsec handling, it is passed through for normal handling by the network. *Id*. This evaluation is automatic, and does not require user intervention, other than as needed to satisfy authentication requirements. *Id.* at 4-5, 30.

362. RFC 2401 describes four diagrams of examples of combinations of SAs according to IPsec. Ex. 1008 (RFC 2401) at 24-26. The legend for the diagrams is as follows:

```
==== = one or more security associations (AH or ESP, transport
       or tunnel)
---- = connectivity (or if so labelled, administrative boundary)
Hx   = host x
SGx  = security gateway x
X*   = X supports IPsec
```

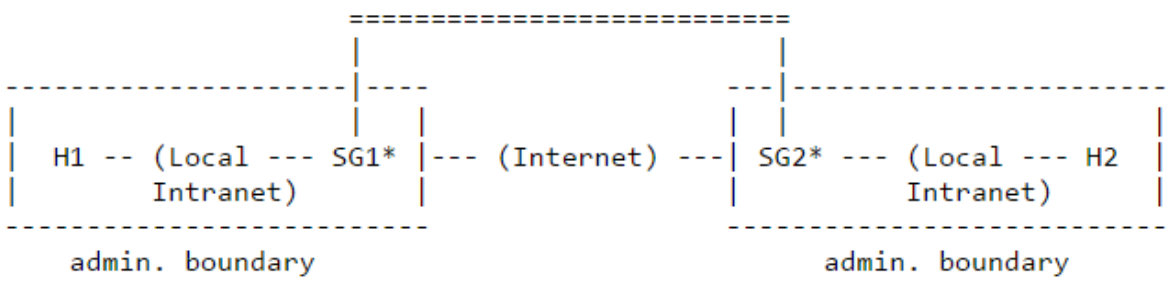
Ex. 1008 (RFC 2401) at 24. The SAs can be AH or ESP. *Id.*

363. RFC 2401 explains how to create an encrypted link that extends from a first network device to the second network device. RFC 2401 specifically describes four communications scenarios: (1) direct encrypted link between two hosts on the internet; (2) virtual encrypted link between two hosts, each in a private network, via gateway devices on the internet, where the connection between gateway devices is encrypted; (3) virtual encrypted link between two hosts, each in a private network, via gateway devices on the internet, where the connection

between gateway devices is encrypted and the virtual connection between hosts is further encrypted; (4) virtual encrypted link between a host on the internet and another host in a private network, via a gateway device on the internet, where the connection between the host on the internet and the gateway device is encrypted and the virtual connection between hosts may be further encrypted:

```
Case 1.  The case of providing end-to-end security between 2 hosts
     across the Internet (or an Intranet).

          ====================================
          |                                  |
        H1* ------ (Inter/Intranet) ------ H2*

     Note that either transport or tunnel mode can be selected by the
     hosts.  So the headers in a packet between H1 and H2 could look
     like any of the following:

          Transport                Tunnel
     -----------------        ---------------------
     1. [IP1][AH][upper]      4. [IP2][AH][IP1][upper]
     2. [IP1][ESP][upper]     5. [IP2][ESP][IP1][upper]
     3. [IP1][AH][ESP][upper]


     Note that there is no requirement to support general nesting,
     but in transport mode, both AH and ESP can be applied to the
     packet.  In this event, the SA establishment procedure MUST
     ensure that first ESP, then AH are applied to the packet.
```

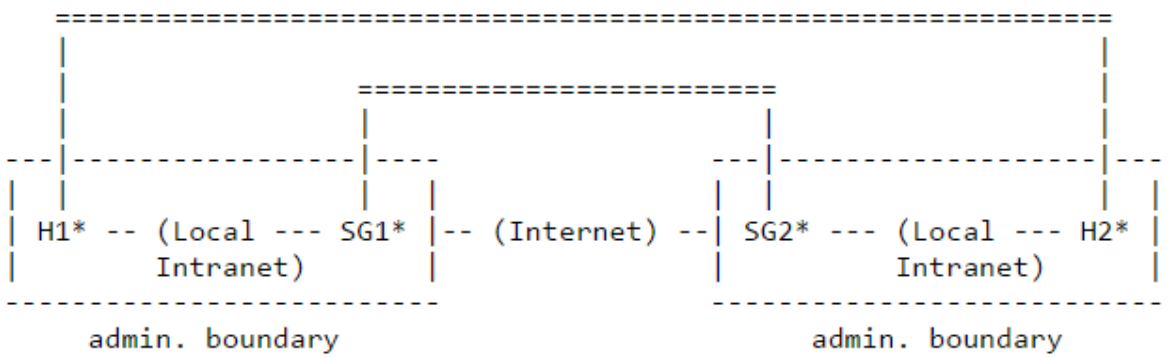
Case 2. This case illustrates simple virtual private networks support.



Only tunnel mode is required here. So the headers in a packet between SG1 and SG2 could look like either of the following:

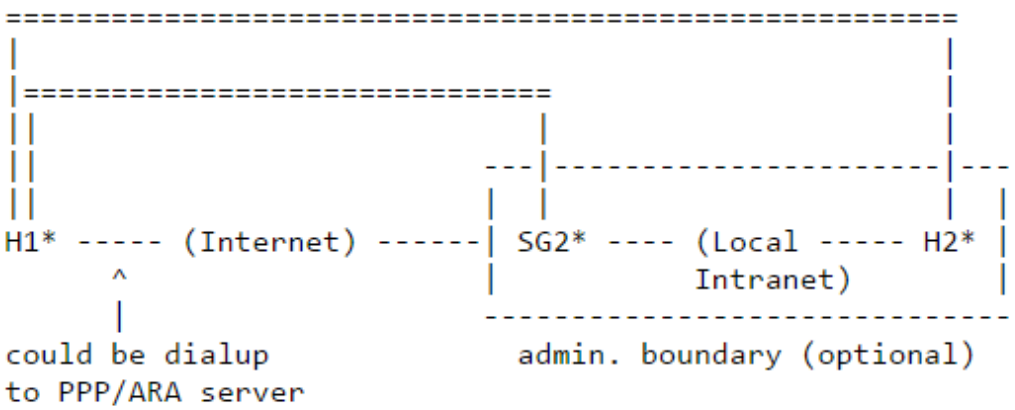
	Tunnel
4.	[IP2][AH][IP1][upper]
5.	[IP2][ESP][IP1][upper]

Case 3. This case combines cases 1 and 2, adding end-to-end security between the sending and receiving hosts. It imposes no new requirements on the hosts or security gateways, other than a requirement for a security gateway to be configurable to pass IPsec traffic (including ISAKMP traffic) for hosts behind it.



Case 4. This covers the situation where a remote host (H1) uses the Internet to reach an organization's firewall (SG2) and to then gain access to some server or other machine (H2). The remote host could be a mobile host (H1) dialing up to a local PPP/ARA server (not shown) on the Internet and then crossing the Internet to the home organization's firewall (SG2), etc. The

details of support for this case, (how H1 locates SG2, authenticates it, and verifies its authorization to represent H2) are discussed in Section 4.6.3, "Locating a Security Gateway".



Only tunnel mode is required between H1 and SG2. So the choices for the SA between H1 and SG2 would be one of the ones in case 2. The choices for the SA between H1 and H2 would be one of the ones in case 1.

Note that in this case, the sender MUST apply the transport header before the tunnel header. Therefore the management interface to the IPsec implementation MUST support configuration of the SPD and SAD to ensure this ordering of IPsec header application.

As noted above, support for additional combinations of AH and ESP is optional. Use of other, optional combinations may adversely affect interoperability.

Ex. 1008 (RFC 2401) at 24-25.

364. As can be seen in the illustrations for cases 3 and 4 above from RFC 2401, an end-to-end encrypted connection between hosts H1 and H2 is established when one or both are on private networks and must pass their communications through one or two security gateways. Ex. 1008 (RFC 2401) at 25-26.

2. Implementing Aventail Using the RFC 2401 Protocol

365. I have been asked to consider whether it would have been obvious to one of ordinary skill in the art circa 2000 to combine Aventail with IETF RFC 2401. It is my opinion that this combination would have been obvious, for the reasons explained below.

366. First, I note that RFC 2401 and Aventail are directed to the same general technical problem – namely, securing network communications.

367. Aventail likewise provides a system where an encrypted communications link is created between a first network device on the public network and a gateway device at the boundary of a public network segment and a private network segment of an organization (see Ex. 1009 (ACAG) at 11-12, 72-73). In Aventail, the private and public network segments of the organization can be further protected by a firewall router, which is used to prevent unauthorized access from the public network. Ex. 1009 (ACAG) at 6:

> 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.

368. Thus, while Aventail provides a secure and encrypted connection between a client running Aventail Connect and the target device on the private network, Aventail only discusses using encryption from the Aventail Connect

client to the Aventail Extranet Server (and vice-versa), as opposed to end-to-end encryption to the target device.

369. A person of ordinary skill in the art would have understood that it was desirable to use end-to-end encryption –as described in RFC 2401 – to protect communications to a private network. For example, U.S. Patent Number 6,408,336 (to Schneider et al., filed March 4, 1998) is also directed to allowing a network device to securely communicate with other devices on a remote network by way of a proxy server. The '336 patent teaches an "end-to-end" encryption scheme, Ex. 1020 (336 patent) at 17:42 – 18:60, because internal networks are not necessarily secure. Ex. 1020 (336 patent) at 5:46-49, 17: 47-51. Similarly, U.S. Patent 5,633,934 (to Hember, filed June 26, 1996) shows a way to encrypt communications on a local area network so as to keep information within user organizations secure. Ex. 1021 (934 patent) at 1:25-27.

370. U.S. Patent Number 6,557,037 (to Provino, filed May 29,1998) provides further evidence that a person of ordinary skill in the art would have considered this a routine design choice, as confirmed by the following sentence in the patent:

> by the destination device. The source and destination devices may themselves perform the encryption and decryption, respectively, or the encryption and decryption may be performed by other devices prior to the message packets being transferred over the Internet.

Ex. 1024 (037 patent) at 2:32-36.

371. Indeed, performing end-to-end encryption between a first network device and a second network device would have been a routine design choice for one of ordinary skill in the art. Many widely used programs in 2000 that communicated over a network made use of end-to-end encryption, including the SSH program and web browsers that used SSL to securely communicate with a web server (for example, a web server that provided financial information such as a bank).

372. One of ordinary skill in the art would have recognized that an end-to-end encryption mechanism – such as RFC 2401's – was desirable and would have been motivated to combine RFC 2401 with Aventail for several reasons.

373. First, as I have noted above, both of these references are directed to providing secure communications between devices that wish to communicate over a network through a gateway.

374. Second, one would have been motivated to combine Aventail and RFC 2401 because they are each layered software components that are designed to be compatible with other software components handling network communication. For example, Aventail explains that Aventail Connect is a layer service provider (LSP) that sits between WinSock and the TCP/IP stack, and that multiple layered service providers can be installed in addition to Aventail Connect that can act to change (encrypt, compress, etc.) data being sent by an application. Ex. 1009

(ACAG) at 9 ("Multiple LSP applications can be installed at the LSP level.").

Aventail describes a layered service provider as:

> A program that is installed just below WinSock 2.0, allowing two-way communication between the WinSock 2.0-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.

Ex. 1009 (ACAG) at 109.

Thus, Aventail expressly informs the reader that one could combine Aventail with other layered service providers (such as RFC 2401).

375. RFC 2401 likewise a layered piece of software that is expressly designed to interoperate with other layered pieces of software at the IP level. As RFC 2401 explains, "IPSec is designed to provide interoperable, high quality, cryptographically-based security for IPv4 and IPv6. […] These services are provided at the IP layer, offering protection for IP and/or upper layer protocols." RFC 2401 at 4. One of ordinary skill in the art would thus understand that both Aventail and RFC 2401 describe layered software packages that are designed to be interoperable with each other and that expressly allow for the possibility of being combined.

376. Yet another reason why one of ordinary skill in the art would be motivated to combine RFC 2401 and Aventail is because RFC 2401 expressly

notes that network security and management is outside its scope, but necessary to ensure security. As RFC 2401 explains:

> The suite of IPsec protocols and associated default algorithms are designed to provide high quality security for Internet traffic. However, the security offered by use of these protocols ultimately depends on the quality of the their implementation, which is outside the scope of this set of standards. Moreover, the security of a computer system or network is a function of many factors, including personnel, physical, procedural, compromising emanations, and computer security practices. Thus IPsec is only one part of an overall system security architecture.
>
> Finally, the security afforded by the use of IPsec is critically dependent on many aspects of the operating environment in which the IPsec implementation executes. For example, defects in OS security, poor quality of random number sources, sloppy system management protocols and practices, etc. can all degrade the security provided by IPsec. As above, none of these environmental attributes are within the scope of this or other IPsec standards.

Ex. 1008 (RFC 2401) at 5 (emphasis added).

377. As one of ordinary skill in the art would understand from reading this disclosure, RFC 2401 is explaining that proper system and network management – which would include requiring authentication to access a remote network, like a SOCKS v5 server such as Aventail provides – is necessary to ensure that IPSec properly protects communications. Thus, one of ordinary skill would have been motivated to combine RFC 2401 with Aventail because Aventail offered just such an environment. Indeed, persons in the field were in fact studying using SOCKS v5 with IPsec for the purposes of providing VPN functionality. See, for example, Manuel Günter, Virtual Private Networks over the Internet:

> **SOCKS v5 and SSL.** SOCKS v5 was originally approved by the IETF as a standard protocol for authenticated firewall traversal. When combined with the Secure Socket Layer (SSL) it provides the foundation for building highly secure VPNs that are compatible with any firewall. SOCKS v5 strength is access control.
>
> . . .
>
> SOCKS v5 and SSL can interoperate on top of IPv4, IPsec, PPTP, L2TP or any other lower level VPN protocol.

Ex. 1025 (Gunter) at 4.

378. Therefore, it is my opinion that a person of ordinary skill in the art would have been motivated to combine Aventail with RFC 2401, leading to an implementation of Aventail where the entire path from a first network device (the client running Aventail Connect) to a second network device (the target computer on the remote private network) is encrypted.

379. A person of ordinary skill in the art would have been able to implement the end-to-end encryption of described in RFC 2401 in connection with the Aventail system. One of ordinary skill in the art would have been familiar the techniques described in RFC 2401 and would have known how to implement networks that utilized those techniques.

380. Indeed, RFC 2401 itself provides guidance on how to implement its end-to-end encryption in existing systems. For example, RFC 2401 explains that its end-to-end encryption technique "imposes no new requirements on the hosts or security gateways, other than a requirement for a security gateway to be

configurable to pass IPsec traffic…" Ex. 1008 (RFC 2401) at 25. This disclosure demonstrates that IPsec would be compatible with Aventail, because Aventail explains that its ExtraNet servers are capable of directly forwarding encrypted network traffic. *See* Ex. 1009 (ACAG) at 63; *see also id.* at 60.

381. There are several ways that a person of skill in the art could have incorporated IPsec into the SOCKS operation of Aventail.

382. For example, when a client requested a connection to a remote host on a network served by a SOCKS server, the SOCKS server (e.g., Aventail ExtraNet server) could provide a SOCKS response that included the network address of the remote host as part of the SOCKS response. That option would have been considered by a person of ordinary skill in the art because the SOCKS protocol already defines SOCKS responses as including the network address that the client computer should use for communications. *See, e.g.*, RFC 1928 at 6 ("In the reply to a CONNECT, BND.PORT contains the port number that the server assigned to connect to the target host, while BND.ADDR contains the associated IP address"). The requesting client and remote host could then establish an IPsec tunnel—just as is disclosed by RFC 2401. Ex. 1008 (RFC 2401) at 26; *see also id*. at 24-26. In this scenario, RFC 2401 explains that the intermediary server (e.g., Aventail Extranet server) would simply forward the encrypted IPsec traffic, Ex. 1008 (RFC 2401) at 25, capability already present in the Aventail scheme. *See* Ex. 1009

(ACAG) at 63; *see also id.* at 60. The client computer and remote host could then exchange encryption parameters (certificates, keys, encryption algorithms). This process could be initiated by the client application trying to connect to the remote host and could be viewed by the client application as a DNS lookup followed by a TCP handshake. Thus the parameters provided to the client to establish the connection would be for a connection that was encrypted from end-to-end. From the perspective of a person of ordinary skill this implementation would have been obvious to try, would have used Aventail and RFC 2401 in a predictable way, and would have routine effort to implement.

3. Incorporating RFC 2401 into the Beser System

383. Beser explains that network traffic in an IP tunnel is ordinarily encrypted to increase the security of the tunnel. For example, it explains:

> One method of thwarting the hacker is to establish a Virtual Private Network ("VPN") by initiating a tunneling connection between edge routers on the public network. For example, tunneling packets between two end-points over a public network is accomplished by encapsulating the IP packet to be tunneled within the payload field for another packet that is transmitted on the public network. The tunneled IP packets, however, may need to be encrypted before the encapsulation in order to hide the source IP address.

Ex. 1007 (Beser) at 2:6-14.

384. Beser explains that traffic within an IP tunnel may be encrypted using standard techniques such as, for example, the "IPsec" protocol. Ex. 1007 (Beser)

at 1:54-55. Beser also uses encryption in establishing its IP tunnels to protect the unique identifier. Ex. 1007 (Beser) at 11:22-25, 18:2-5, 20:11-14.

385. Beser refers to the IPsec protocol (described in RFC 2401) to explain how encryption is typically incorporated into IP tunnels. Ex. 1007 (Beser) at 1:54-56. A person of ordinary skill in the art would have considered Beser in conjunction with RFC 2401 because Beser expressly refers to the IPsec protocol as a standard encryption technique typically used in IP tunnels. Ex. 1007 (Beser) at 1:54-56.

386. Beser's suggestion that encryption techniques such as IPsec can be used to encrypt traffic in IP tunnels is consistent with the guidance in the IPsec standard, RFC 2401 (Ex. 1008). For example, RFC 2401 explains:

> IPsec allows the user (or system administrator) to control the granularity at which a security service is offered. For example, one can create a single encrypted tunnel to carry all the traffic between two security gateways or a separate encrypted tunnel can be created for each TCP connection between each pair of hosts communicating across these gateways. IPsec management must incorporate facilities

Ex. 1008 (RFC 2401) at 7.

387. Beser identifies certain situations in which a person might choose not to use encryption in an IP tunnel because of practical concerns related to transmission of high volumes of data (*e.g.*, for VOIP and multimedia settings) As Beser explains:

> mation to decrypt the message. Moreover, encryption at the source and decryption at the destination may be infeasible for certain data formats. For example, streaming data flows, such as multimedia or Voice-over-Internet-Protocol ("VoIP"), may require a great deal of computing power to encrypt or decrypt the IP packets on the fly. The increased strain on computer power may result in jitter, delay, or the loss of some packets. The expense of added computer power might also dampen the customer's desire to invest in VoIP equipment.

Ex. 1007 (Beser) at 1:58-67.

388. The practical concerns associated with encryption that Beser identifies are limited to high volume data transfer situations. A person of ordinary skill in the art reading Beser in February 2000 would recognize that using encryption in connection with the Beser tunneling scheme would not present any practical concerns in situations other than the high data volume scenarios, such as communications carrying video data.

389. A person of ordinary skill in the art reading Beser in February 2000 would also have understood that encryption should ordinarily be used even in high data volume applications, if possible. Beser only warns that it *may* not be possible to encrypt every packet and maintain transmission quality due to computer power limitations (*e.g.*, during times of high network traffic). Ex. 1007 (Beser) at 1:62-67, 2:15-17. Beser refers to the use of encryption in IP tunneling schemes as a conventional technique that is ordinarily used. *See* Ex. 1007 (Beser) at 1:54-56, 2:12-14.

390. A person of ordinary skill in the art would recognize that the concerns expressed in Beser in connection with encryption of high volumes of network traffic can be easily resolved by simply using more powerful equipment. Alternatively, systems may be configured to transmit multimedia content using less data (*e.g.*, by using lower resolution video).

391. Beser also explains that the transmission of a request containing a unique identifier "may require encryption or authentication to ensure that the unique identifier cannot be read on the public network." Ex. 1007 (Beser) at 11:22-25; *see also* 18:2-5, 20:11-14. In other words, the trusted-third-party network device can be configured to require that requests be encrypted and clients be authenticated before it will process a request. *See id.*

392. Although Beser does not specifically describe the authentication process, many authentication techniques were well known in February of 2000. *See, e.g.*, ¶140 above. The decision of which authentication technique to use would have been a simple design choice.

393. To the extent that Beser does not teach the details of an encrypted configuration, a person of ordinary skill in the art would have found it obvious in February 2000 to modify Beser to incorporate encryption techniques known in the art. This is because Beser describes its process as able to provide additional security to existing security techniques, such as VPNs. *See* Ex. 1007 (Beser) at

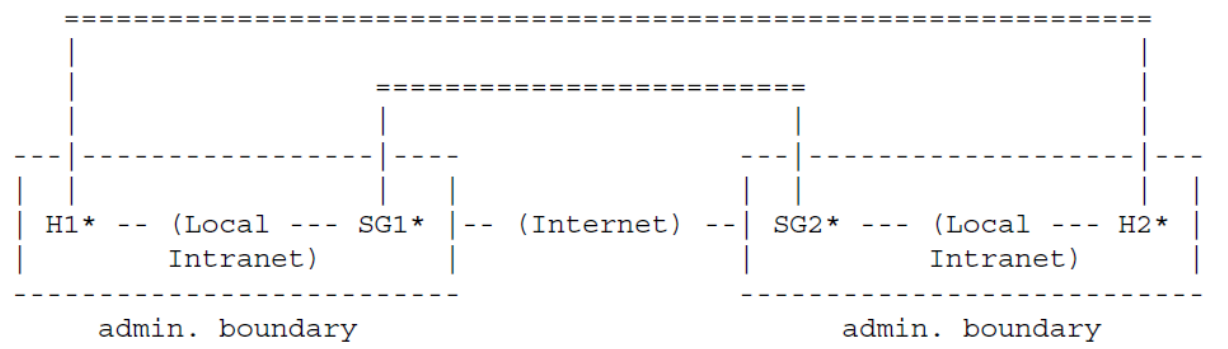
2:1-13. Beser also explains that its methods and systems may be used to connect devices on separate Local Area Networks via the Internet, Ex. 1007 (Beser) at 4:5-29, or to establish communications across corporate offices, Ex. 1007 (Beser) at 10:55-67.

394. Beser explains that its processes are flexible and many different configurations of systems implementing its processes are possible. Ex. 1007 (Beser) at 5:3-14.

395. Beser refers to the IPsec protocol to explain how encryption is conventionally incorporated into IP tunneling schemes. Ex. 1007 (Beser) at 1:54-56. A person of ordinary skill in the art would immediately recognize that, under the IPsec protocol, communications within an IP tunnel (*e.g.*, between a first and second network device or between the two end devices) to initiate the tunnel and following establishment of the IP tunnel will be encrypted. *See* Ex. 1007 (Beser) at 1:56-2:15.

396. One of the examples described in RFC 2401 (*i.e.*, "case 3") shows encryption between two security gateways, and another encrypted tunnel between two end devices:

Case 3. This case combines cases 1 and 2, adding end-to-end security between the sending and receiving hosts. It imposes no new requirements on the hosts or security gateways, other than a requirement for a security gateway to be configurable to pass IPsec traffic (including ISAKMP traffic) for hosts behind it.



Ex. 1008 (RFC 2401) at 25.

397. "Case 3" described in RFC 2401 is the precise network design shown in Figure 1 of Beser (*i.e.*, a configuration in which edge routers are connected to private networks and communicate over a public network). *Compare* Ex. 1007 (Beser) Fig, 1 *and* Ex. 1008 (RFC 2401) at 25.

398. Beser also indicates its IP tunneling schemes are compliant with standards-based processes and techniques and can be implemented using pre-existing equipment and systems. Ex. 1007 (Beser) at 4:55-5:2. A person of ordinary skill in the art would have understood that a standard encryption technique such as IPsec can and should be incorporated into systems implemented according to Beser.

399. A person of ordinary skill in the art would have been motivated to encrypt IP traffic within the IP tunnel of Beser based on the teachings of RFC

2401, which explains how to provide encryption protection for the precise network configuration shown in Figure 1 of Beser. *Compare* Ex. 1007 (Beser), Fig. 1 *and* Ex. 1008 (RFC 2401) at 25. In the combined system, IPsec could provide end-to-end encryption, hiding the data, while the Beser tunnel would provide anonymity over the public network, hiding the true source and destination addresses. The exchange of private addresses for the originating and terminating end devices according to Beser's technique would enable such encrypted communications through the IP tunnel.

400. In this combined system, it would have been obvious to configure the Beser trusted-third-party network device to perform the IPSec key management and distribution functionality described in Sections 4.6 and 4.7 of RFC 2401. Ex. 1008 (RFC 2401) at 26-29. The trusted-third-party network device in the Beser system is a trusted intermediary, and it would have been natural to configure this device to support key management and distribution on behalf of end devices.

401. If the end devices in Beser wish to use the IKE protocol (RFC 2409) to mutually agree on the cryptographic methods and shared session keys to securely communicate with IPsec, Beser's trusted-third party could be used as a certification authority or key-distribution-center to enable mutual authentication of the end devices within the IKE negotiation. *See* ¶140 above.

402. Alternatively, if the end devices in Beser's scheme wish to use a KDC to provide them with shared session keys to securely communicate with IPsec, Beser's trusted-third party could generate such session keys and provide them to the end devices by communicating with one or both of them. *See* ¶138 and ¶140 above.

403. For example, if the connection called for end-to-end encryption, the trusted-third-party network device could encrypt one copy of the session key with the terminating end device's registered key, and another copy with the originating device's registered key. *See* ¶138 above. This would allow the key to be passed to those devices without risk that the intermediary network devices could read the key. Similarly, if the connection called for encryption only between the first and second network devices (security gateways), the trusted-third-party network device could provide a session key directly to those devices.

D. U.S. Patent No. 5,237,566 to Brand (Ex. 1012)

1. Overview of Brand

404. U.S. Patent No. 5,237,566 to Brand ("Brand") (Ex. 1012) was filed on March 30, 1989 and issued on August 17, 1993.

405. Brand concerns a hub network system that maintains full media bandwidth for a plurality of nodes. Ex. 1012 (Brand) at 1:5-8. The system

described in Brand is designed for bus baseband networks. Ex. 1012 (Brand) at 2:2:47-49.

406. As Brand explains, the amount of data that can be transferred over a channel is known as bandwidth. Ex. 1012 (Brand) at 1:26-28. Two types of bandwidth are "baseband" and "broadband." Ex. 1012 (Brand) at 1:28-29. Brand explains:

> systems and baseband systems. In a broadband system, several data terminals share a single communication medium through a frequency-division scheme. In a baseband configuration, the signals are unmodulated, and sharing of a medium requires time division between the nodes. In either system, when more than one data

Ex. 1012 (Brand) at 1:29-34.

407. Brand identifies LocalTalk as an example of a baseband network system. Ex. 1012 (Brand) at 2: 47-51. Ethernet networking technologies also rely on baseband transmission. Ex. 1012 (Brand) at 17:44-46.

408. The system described in Brand results in increased efficiency because it allows each terminal in the network to communicate at the full bandwidth allowed by a connection. Ex. 1012 (Brand) at 3:29-35.

2. Combination of Aventail and Brand

409. I have been asked to consider whether it would have been obvious to one of ordinary skill in the art circa 2000 to combine the methods and systems described in Aventail with Brand's teachings regarding baseband networks,

broadband networks, and modulation. It is my opinion that this combination would have been obvious, for the reasons explained below.

410. Aventail describes communications over networks, but does not detail the characteristics of the physically implementation of those networks. *See, e.g*., Ex. 1009 (ACAG) at 1, 5-12. Brand provides details about the characteristics of the physical implementation of networks. For example, Brand characterizes networks as either broadband or baseband, based on the type of bandwidth used by the network. *See, e.g.,* Ex. 1012 (Brand) at 1:26-29. Brand further explains the characteristics of these two network types. For example, Brand explains that broadband networks use modulation techniques such as frequency division. *See, e.g.,* Ex. 1012 (Brand) at 1:29-31. Brand also explains that baseband networks operate without modulation. *See, e.g.,* Ex. 1012 (Brand) at 1:31-34.

411. In my opinion, a person of ordinary skill in the art implementing Aventail would readily have consulted Brand to implement the physical aspects of the networks required by Aventail. Indeed, a person of ordinary skill in the art would have known to use either of the two basic types of networks disclosed by Brand. *See, e.g.,* Ex. 1012 (Brand) at 1:26-29.

412. Thus, it would have been obvious to implement Aventail using a broadband network. *See, e.g.,* Ex. 1012 (Brand) at 1:26-29. In that implementation, a person of ordinary skill in the art would have known that the

implantation used modulation, because that is a characteristic of broadband networks. *See, e.g.,* Ex. 1012 (Brand) at 1:29-31. In addition, a person of ordinary skill in the art would have known the different types of modulation, such as frequency division modulation. Ex. 1012 (Brand) at 1:29-31.

413. It would also have been obvious to implement Aventail using baseband networks. *See, e.g.,* Ex. 1012 (Brand) at 1:26-29. In that implementation, a person of ordinary skill in the art would have known that the implementation did not use modulation, because that is a characteristic of baseband networks. *See, e.g.,* Ex. 1012 (Brand) at 1:31-34.

3. Combination of Beser and Brand

414. I have been asked to consider whether it would have been obvious to one of ordinary skill in the art circa 2000 to combine the methods and systems described in Beser with Brand's teachings regarding baseband networks. It is my opinion that this combination would have been obvious, for the reasons explained below.

415. Beser explains that the private networks that connect the originating end device and the first network device, and the terminating end device and the second network device, respectively, can be Local Area Networks ("LANs"). Ex. 1007 (Beser) at 4:19-42, Fig. 1.

416. Ethernet and LocalTalk were two types of LAN technology known in 2000. It was also known in 2000 that Ethernet and LocalTalk were baseband networks, which, as Brand explains, are "unmodulated." Ex. 1012 (Brand) at 1:31-34.

417. It would have been obvious to a person of ordinary skill to implement Brand's teaching regarding baseband LAN networks in the Beser system because Beser is designed to be compatible with LAN networks. Ex. 1007 (Beser) at 4:19-42. Thus, Beser expressly informs the reader that one could combine Beser with unmodulated communication channels as those described in Brand.

E. RFC 2543, the Session Initiation Protocol

1. Overview of RFC 2543

418. "Request for Comments: 2543; SIP: Session Initiation Protocol" ("RFC 2543") (Ex. 1013) was published in March 1999. RFC 2543 describes a network-based secure video telephony architecture that supports both audio and video conferences. It explains, for example:

> The Session Initiation Protocol (SIP) is an application-layer control (signaling) protocol for creating, modifying and terminating sessions with one or more participants. These sessions include Internet multimedia conferences, Internet telephone calls and multimedia distribution.

Ex. 1013 (RFC 2543) at 1.

419. RFC 2543 explains that telephony functionality supports audio and video:

> For example, assume that the caller Alice has included the following description in her INVITE request. It includes an audio stream and two bidirectional video streams, using H.261 (payload type 31) and MPEG (payload type 32).

Id. at 137

420. RFC 2543 also explains that its telephony communications can be encrypted:

> 6.19 Encryption
>
> The Encryption general-header field specifies that the content has been encrypted. Section 13 describes the overall SIP security architecture and algorithms. This header field is intended for end-to-end encryption of requests and responses. Requests are encrypted based on the public key belonging to the entity named in the To header field. Responses are encrypted based on the public key conveyed in the Response-Key header field. Note that the public keys themselves may not be used for the encryption. This depends on the particular algorithms used.

Ex. 1013 (RFC 2543) at 54; *see also id.* at 54-55, 104-106.

2. Combining Aventail with RFC 2543

421. Aventail explains that its networking scheme is based on "protocol-independent" encryption techniques, Ex. 1009 (ACAG) at 27, thus disclosing that the scheme was applicable to a wide variety of applications. Similarly, Aventail discloses implementations in which a number of different network-based services are provided by way of encrypted network communication channels. *See, e.g.*, Ex.

1009 (ACAG) at 72 (diagram showing VPN with DNS, WWW, mail, news, WinFrame, and SQL servers).

422. Persons of ordinary skill in the art would have recognized that the telephony functionality taught by RFC 2543 would have been one of the protocols that could be utilized with the protocol-independent multipurpose scheme taught by Aventail. Persons of ordinary skill in the art would have recognized that including network-based telephony services on a single, common communications architecture was both desirable and a conventional design technique. Moreover, a person of ordinary skill in the art would have been motivated to do this, as it would enable organizations to consistently implement and regulate security and access control measures. Implementing support for audiovisual telephony services, such as those described in RFC 2543, within the Aventail architecture also would have been simple and straightforward from a technical perspective. In February of 2000, Aventail in view of RFC 2543 thus would have suggested implementing audiovisual telephony functionality within an encrypted communications scheme, and would have considered doing so to be a routine design choice.

F. Additional Prior Art Combinations

1. Incorporating Beser with RFC 2401 with Brand

423. As I have explained above, it would have been obvious to one of ordinary skill in the art to combine Beser with RFC 2401 and to combine Beser

with Brand. It is also my opinion that it would have been obvious to one of ordinary skill in the art to combine all three of these references together. As I have previously explained, it would have been obvious to incorporate RFC 2401 with Beser, because Beser notes that standard encryption techniques can be used to encrypt network communications and RFC 2401 provides just such a technique. It would have been obvious for one of ordinary skill in the art to incorporate Brand's discussion of network types into this combined system, because Brand is directed to two different types of networks, and one of ordinary skill in the art would have understood that it would have been advantageous for the combined system of Beser and RFC 2401 (offering encrypted end-to-end communication) to work on as many network types as possible.

2. Combining Aventail, RFC 2401, and RFC 2543

424. As I have explained above, it would have been obvious to one of ordinary skill in the art to combine Aventail with RFC 2401 and to combine Aventail with RFC 2543. It is also my opinion that it would have been obvious to one of ordinary skill in the art to combine all three of these references together. As I have previously explained, it would have been obvious to incorporate RFC 2401 with Aventail because, among other things, while Aventail discloses encrypted communications but not end-to-end encryption, persons of skill in the art would have recognized that an end-to-end encryption scheme would have provided

greater security. It would have been obvious for one of ordinary skill in the art to incorporate RFC 2543 into Aventail, because RFC 2543 is directed to adding audio-video telephony with encryption support to networks, and one of ordinary skill in the art would have understood that it would have been advantageous for the combined system of Aventail and RFC 2401 (offering encrypted end-to-end communication) to provide secure audio-video telephony as suggested by RFC 2543. Indeed, RFC 2543 discloses that it can be used with end-to-end encryption. *See, e.g.*, Ex. 1013 (RFC 2543) at 54.

3. Combining Aventail, RFC 2401, and Brand

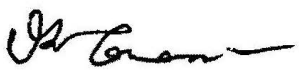
425. As I have explained above, it would have been obvious to one of ordinary skill in the art to combine Aventail with RFC 2401 and to combine Aventail with Brand. It is also my opinion that it would have been obvious to one of ordinary skill in the art to combine all three of these references together. As I have previously explained, it would have been obvious to incorporate RFC 2401 with Aventail, because while Aventail discloses encrypted communications but not end-to-end encryption, persons of skill in the art would have recognized that an end-to-end encryption scheme would have provided greater security. It would have been obvious for one of ordinary skill in the art to incorporate Brand's discussion of network types into this combined system, because Brand is directed to two different types of networks, and one of ordinary skill in the art would have

understood that it would have been advantageous for the combined system of Aventail and RFC 2401 (offering encrypted end-to-end communication) to work on as many network types as possible.

* * *

426. I, Roberto Tamassia, do hereby declare and state, 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, under Section 1001 of Title 18 of the United States Code.

427. Executed on: March 2, 2015



APPENDIX A

Exhibit #	Reference Name
1001	U.S. Patent 8,868,705
1002	U.S. Patent 8,868,705 File History
1003	U.S. Patent 8,850,009
1004	U.S. Patent 8,850,009 File History
1007	U.S. Patent 6,496,867 to Beser
1008	Kent, S., et al., RFC 2401, "Security Architecture for the Internet Protocol" (November 1998)
1009	Aventail Connect v3.01/v2.51 Administrator's Guide (1996-1999)
1010	Aventail Connect v3.01/v2.51 User's Guide (1996-1999)
1011	Aventail ExtraNet Center v3.0 Administrator's Guide (NT and UNIX)
1012	U.S. Patent 5,237,566 to Brand
1013	Handley, M., et al., RFC 2543, SIP: Session Initiation Protocol (March 1999)
1014	RFC 793, DARPA Internet Program Protocol Specification (September 1991)
1015	U.S. Patent Number 6,430,176 to Christie
1016	U.S. Patent Number 6,930,998 to Sylvain
1017	Patent Owner Preliminary Response, IPR2014-00237, Paper 12 (March 6, 2014)
1018	Leech, M., et al., RFC 1928, SOCKS Protocol Version 5 (March 1996)
1019	Microsoft Computer Dictionary (4th Ed.)
1020	U.S. Patent Number 6,408,336 to Schneider
1021	U.S. Patent Number 5,633,934 to Hember
1022	Exhibit E3 of Reexamination 95/001,697 - Declaration of James Chester
1023	Exhibit E1 of Reexamination 95/001,697 - Declaration of Chris Hopen,

Exhibit #	Reference Name
1024	U.S. Patent 6,557,037 to Provino
1025	Gunter, M., "Virtual Private Networks Over the Internet" (1998)
1026	Patent Owner Preliminary Response, IPR2014-00404, Paper 9 (May 19, 2014)
1027	von Sommering, S., Space Multiplexed-Electrochemical Telegraph
1028	Licklider, J.C.R., Memorandum for Members and Affiliates of the Intergalactic Computer Network (December 11, 2001)
1029	BBN Report No. 1822, Interface Message Processor (January 1976)
1030	Koblas, D., et al., "SOCKS-Usenix Unix Security Symposium III"
1031	Windows NT for Dummies
1032	Mockapetris, P., RFC 882, "Domain Names – Concepts and Facilities" (November 1983)
1033	Mockapetris, P., RFC 883, "Domain Names – Implementation and Specification" (November 1983)
1034	Mockapetris, P., RFC 1034, "Domain Names – Concepts and Facilities" (November 1987)
1035	Mockapetris, P. RFC 1035, "Domain Names – Implemenation and Specification" (November 1987)
1036	Bradner, S., RFC 2026, "The Internet Standards Process – Revision 3" (October 1996)
1037	Rosen, E., et al., RFC 2547, "BGP/MPLS VPNs" (March 1999)
1038	W3C and WAP Forum Establish Formal Liason Relationship (December 8, 1999)
1039	Wireless Application Protocol (WAP) Architecture Specification (April 30, 1998)
1040	H.323 ITU-T Recommendation (February, 1998)
1041	Kiuchi, T., et al., "C-HTTP – The Development of a Secure, Closed HTTP-Based Network on the Internet" (IEEE 1996) LOC Stamped
1042	VirnetX's Opening Claim Construction brief, VirnetX, Inc. v. Cicso Systems, Inc., et al., Civil Action No. 6:10-cv-417 (EDTX) (November 4, 20111)
1043	Exhibit E2 of Reexamination 95/001,682 - Declaration of Michael

Exhibit #	Reference Name
	Fratto
1044	U.S. Patent 5,898,830 to Wesinger
1045	Steiner, J., et al., "Kerberos: An Authentication Service for Open Network Systems" (January 12, 1988)
1046	Harkins, D., et al., RFC 2409, "The Internet Key Exchange (IKE) (November 1998)
1047	Maughan, D., et al., "Internet Security Association and Key Management Protocol (ISAKMP)" (November 1998)
1048	Data-Over-Cable Interface Specifications (DOCIS), Radio Frequency Interface Specification (March 26, 1997)