As set forth in Ex.1009 ¶¶ 29-39 and Appendix C, Kiuchi discloses all of the limitations of claim 1.

Step (1) of Claim 1 - When the client-side proxy (i.e., the "client computer") receives the HTTP request from the user agent, it generates (and transmits) a DNS request that is sent to the C-HTTP name server in the form of a C-HTTP name service request to ask the C-HTTP name server whether it can communicate with the specified host. (Ex.1002 at 65, sec. 2.3(2)). The C-HTTP name service request is a "DNS request" because it is a communication that contains a domain name (i.e., the hostname from the URL in the HTTP request) and requests an IP address for the domain name. Because the domain name sent in the DNS request is the hostname given to the server-side proxy, the domain name is literally "associated with the target computer" as required by the claim. Therefore, the DNS request "requests an IP address corresponding to a domain name associated with the target computer" as required by step (1). The client-side proxy is a "client computer" because it is a computer from which the DNS request is generated and transmitted.

Step (2) of Claim 1 - Upon receiving the C-HTTP name service request (i.e., DNS request) from the client-side proxy, the C-HTTP name server "examines whether the requested server-side proxy is registered in the closed network." (Ex.1002 at 65, sec. 2.3(2)). The C-HTTP name server determines whether the DNS request transmitted by the client-side proxy in step (1) is requesting access to

a secure web site based on whether the requested server-side proxy is registered in the closed network. The C-HTTP name server determines that access is being requested to a secure web site being proxied by the server-side proxy (i.e., that access is being requested to a "secure web site"/"secure target web site") and sends a C-HTTP name service response to the client-side proxy containing "the IP address and public key of the server-side proxy and both request and response Nonce values" (Ex.1002 at 65, sec. 2.3(2)), if the requested server-side proxy is registered in the closed network. The C-HTTP name server determines that access is not being requested to a secure web site and sends "a status code which indicates an error," if the requested server-side proxy is not registered in the closed network. (Ex.1002 at 65, sec. 2.3(2)-(3)). Thus, step (2) of claim 1 is satisfied by the determination made by the C-HTTP name server.

Additionally, step (2) of claim 1 is satisfied by a determination made by the client-side proxy based on the type of response received from the C-HTTP name server. In particular, the client-side proxy determines that access is being requested to a secure web site being proxied by the server-side proxy (i.e., that access is being requested to a "secure target web site"), only if a C-HTTP name service response is returned, and determines that access is not being requested to a secure web site, if the response is an error status status.

Step (3) of Claim 1 - As discussed above for step (2), if the C-HTTP name server determines that the requested server-side proxy is registered in the closed network and therefore determines that access is being requested to a secure target web site, the C-HTTP name server sends a C-HTTP name service response to the client-side proxy. The sending of the C-HTTP name service response by the C-HTTP name server to the client-side proxy constitutes "automatically initiating the VPN" within the context of the claim because the C-HTTP name service response causes the client-side proxy to send a request for connection to the server-side proxy. (Ex.1002 at 65, sec. 2.3(3)). The '135 Patent provides, as one example of automatically initiating the VPN, transmission of a message requesting that a VPN be created (see Ex.1001 at 38:30-33). The C-HTTP name service response is analogous because it is a message that causes the VPN to be created. Therefore, step (3) of claim 1 is satisfied by sending of the C-HTTP name service response message to the client-side proxy in response to determining that the DNS request is requesting access to a secure target web site.

Additionally, step (3) of claim 1 is satisfied by actions taken by the client-side proxy to automatically initiate the VPN. In response to receiving a C-HTTP name service response (which the client-side proxy uses to determine that the DNS request is requesting access to a secure target web site), the client-side proxy "sends a request for connection to the server-side proxy, which is encrypted using

the server-side proxy's public key and contains the client-side proxy's IP address, hostname, request Nonce value and symmetric data exchange key for request encryption." (Ex.1002 at 65, section 2.3(3)). This connection request sent by the client-side proxy also meets step (3) of claim 1.

Ground 2. Claim 3 is Anticipated by Kiuchi

As set forth in Ex.1009 ¶ 40 and Appendix C, Kiuchi also discloses all of the limitations of claim 3. In response to determining that the requested server-side proxy is not registered in the closed network (which indicates that the DNS request is not requesting access to a secure target web site), the C-HTTP name server returns to the client-side proxy "a status code which indicates an error." (Ex.1002 at 65, sec. 2.3(2)). In turn, "[i]f the client-side proxy receives an error status, then it performs DNS lookup, behaving like an ordinary HTTP/1.0 proxy." (Ex.1002 at 65, sec. 2.3(2)). It is well-known that such a DNS lookup involves sending a request to a DNS server and receiving an IP address back from the DNS server. (Ex.1010 at 70 et seq.). In this way, the domain name is resolved and the IP address is returned to the client-side proxy, specifically by the client-side proxy sending a lookup request to the conventional DNS server which resolves the domain name and returns the IP address to the client-side proxy. Thus, Kiuchi teaches the limitations of claim 3.

Ground 3. Claim 7 is Anticipated by Kiuchi

As set forth in Ex.1009 ¶¶ 41-43 and Appendix C, Kiuchi also discloses all of the limitations of claim 7 in at least the following two ways:

Gatekeeper Computer Functions Implemented in C-HTTP Name Server

The '135 Patent makes clear that the gatekeeper can be implemented as a function within the DNS server (see Ex.1001 at 38:53-55). As discussed under Ground 1, the C-HTTP name server is a DNS server that performs the step of "automatically initiating the VPN" by sending the C-HTTP name service response to the client-side proxy. In this context, the C-HTTP name server also performs the functions of a gatekeeper computer because it allocates VPN resources, e.g., it generates and provides the request and response Nonce values and returns the public key of the server-side proxy and the request and response Nonce values to the client-side proxy. (Ex.1002 at 65, sec. 2.2).

Gatekeeper Computer Functions Implemented in Server-Side Proxy

Additionally, as discussed under Ground 1, Kiuchi's client-side proxy performs the step of "automatically initiating the VPN" by sending a request for connection to the server-side proxy. In this context, the server-side proxy also performs the functions of a gatekeeper computer because it allocates VPN resources such as a Connection ID and a second symmetric data exchange key that are used in establishing a secure connection between the client-side proxy and the server-side proxy. (see Ex.1002 at 66, sec. 2.3(5)). Under the broadest reasonable

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.