

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

CISCO SYSTEMS, INC.,
Petitioner,

v.

ORCKIT CORPORATION,
Patent Owner.

Case IPR2023-00554
Patent 10,652,111

EXHIBIT 2025

DECLARATION OF MIGUEL GOMEZ

TABLE OF CONTENTS

	<u>Page</u>
I. INTRODUCTION	1
II. QUALIFICATIONS.....	1
III. BASES OF OPINIONS	9
IV. APPLICABLE LEGAL STANDARDS.....	9
A. Ordinary Skill in the Art	9
B. Claim Construction	11
C. Obviousness (35 U.S.C. § 103).....	11
V. OVERVIEW OF THE '111 PATENT AND THE STATE OF THE ART	15
A. Technical Background of the '111 Patent.....	15
B. Petitioner's Cited References	17
1. Lin (Ex. 1005)	17
2. Shieh (Ex. 1006).....	19
3. Swenson (Ex. 1007)	19
C. Level of Ordinary Skill	20
VI. CLAIM CONSTRUCTION.....	21
VII. NO COMBINATION OF THE CITED PRIOR ART REFERENCES TEACHES OR SUGGESTS EVERY LIMITATION OF THE CLAIMS.....	21
A. Lin and Swenson do not render obvious claim 1.....	21

1.	Lin and Swenson do not disclose or render obvious the claimed controller under either Patent Owner’s or Petitioner’s proposed constructions.....	21
a.	Lin does not disclose or render obvious a controller configured to perform or capable of controlling DPI.....	22
b.	Swenson does not disclose or render obvious a controller configured to perform or capable of controlling DPI.....	26
c.	A POSITA would not have combined Swenson’s controller with Lin’s or adapted such a combination to practice the claimed controller in Lin’s system in any event.....	33
d.	Lin does not disclose sending the packet to the second entity “responsive to the packet not satisfying the criterion” or sending the packet to an entity other than the second entity “responsive to the packet satisfying the criterion.”	37
B.	Shieh and Swenson do not render obvious claim 1	41
1.	Shieh does not disclose the claimed “packet-applicable criterion” responsive to which a packet is sent either to a second entity or to another entity that is different from the second entity.....	41
C.	Claim 3: The prior art does not disclose sending the packet to an entity that is other than the second entity <i>and</i> to the <i>controller</i>	44
D.	Claims 4 and 5: The prior art does not disclose sending the packet to an entity that is other than the second entity <i>and</i> to the <i>controller</i>	46
E.	Claim 6: The prior art does not disclose storing <i>the packet</i> by the controller in a memory	46

F.	Claim 7: The prior art does not disclose sending “a portion of the packet” as claimed	49
G.	Claim 16: The prior art does not disclose “the packet comprises distinct header and payload fields, the header comprises one or more flag bits, and wherein the packet-applicable criterion is that one or more of the flag bits is set”	50
H.	Claim 30: The prior art does not disclose “receiving...one or more additional packets” as claimed	51
VIII.	CONCLUSION	54

I. INTRODUCTION

1. My name is Miguel Gomez. I have been retained as an expert witness to provide my independent opinion in regards with matters at issue in the inter partes review of U.S. 10,652,111 (“the '111 Patent”) in the IPR2023-00554 proceeding. I have been retained by Orckit Corporation (“Orckit”), the Patent Owner, in the above proceedings. Petitioner in this case is Cisco Systems, Inc. (“Cisco”).

2. Unless otherwise noted, the statements made herein are based on my personal knowledge, and if called to testify about this declaration, I could and would do so competently and truthfully.

3. My CV is being submitted herewith as Exhibit 2026 to this proceeding and is summarized in Section II, *infra*.

4. I am not a legal expert and offer no opinions on the law. However, I have been informed by counsel of the various legal standards that apply, and I have applied those standards in arriving at my conclusions.

5. The opinions I set forth herein are my own and are based on my education, experience, training, and skill I have accumulated.

II. QUALIFICATIONS

6. I received a Bachelor of Science in Electrical Engineering degree from Yale University in 1983. I have forty years of experience in the design and

development of hardware and software technology used in computer systems, communications systems, networking and storage infrastructure, and database systems. My experience includes extensive knowledge of computer operating systems, programming languages, and mobile and embedded applications. I am also highly skilled in the use of microelectronics simulation software, ASIC and FPGA coding, in hardware design languages, including Verilog and VHDL. I'm also skilled at programming for embedded systems and applications in C, C++ and C#.

7. From August 2006 to March 2009, I was VP of engineering for ActSolar. ActSolar was a solar power converter and instrumentation company where I developed the system architecture that defined the hardware and software for a solar inverter and a data gathering and control sub-system. I wrote a control program in the C programming language to manage power conversion for the solar inverter. I also managed a small team of engineers that built the software defined radio responsible for retrieving solar panel data. The data gatherer utilized the IEEE 802.15.4 low-rate wireless personal area network (WPAN) standard to retrieve power and temperature readings from each solar panel and aggregate that data at a server. This information was used to maximize the solar array output.

8. In 2006, ActSolar was sold to National Semiconductor. The transfer of technology included the software radio for data collection, the inverter and a variant of the power converter hardware that performed shading compensation.

9. From September 2004 to August 2005, I was a consultant for BridgeWave Inc. I built a 10 Gbps Ethernet test platform to facilitate manufacturing tests of BridgeWave's high speed wireless network system. This system replaced the Smart Bits Ethernet test analyzer used in the manufacturing test environment. I also built a test generator, specifically an Ethernet packet generator, which generated packets of various sizes, address spaces, payload and checksums. I built a test receiver and analyzer which detected various failure mechanisms, maintained statistics and reported them to a user interface written in Lab View. A Xilinx Vertex II FPGA was employed in the first version of the test platform and a Xilinx Spartan 3e in the second version.

10. From September 2003 to March 2004, I was a consultant for PA Consulting Group. I provided hardware evaluation services for corporate mergers and acquisitions. These services included systems and hardware reviews in the following areas: (1) Analysis of system performance to the cost of the hardware implementation; (2) Review of the VHDL and Verilog code implementations, documentation and development strategies; (3) VHDL and Verilog code

management including compilation processes, simulation, test coverage, bug tracking and source code control; (4) FPGA and ASIC tool chain management; (5) Circuit board design and layout for production environments; (6) Circuit board certification testing for FCC, UL, and Environmental tests; (7) Review of production line management including assembly and test processes; and (8) Review of hardware code for copyright or license violations.

11. From January 2003 to May 2003, I was a consultant for Santel Networks. I developed an optical duo-binary (ODB) encoder. Operating at 10.7 Giga bits per second (GBps), ODB reduces the spectral spread to $\frac{1}{2}$ of the ordinary NRZ format. This format allows high spectral efficiency over long distances by reducing nonlinear penalties due to narrower channel spacing and allows a higher bit rate or increases transmission distances. The project was implemented in a Xilinx FPGA running at 667 Mhz with a 10.7 GHz Mux/Demux feeding 32 LVDS streams to the FPGA. The ODB behavior was simulated in a C-coded simulation. The circuit board's highest clock speed was 10.7 Gigahertz and was modeled in HSPICE. Board level simulation was performed using NCVerilog. Several patents were applied for and the board was shown at the Optical Fiber Communication Conference & Exposition in March of 2003.

12. From March 2001 to December 2002, I was Director of Hardware, Content Networking Division for Extreme Networks, Inc. I managed a team of hardware and software engineers that developed and maintained a Layer 2-5 Content Addressable Switch. The switch was capable of L2, L3, L4 switching as well as L5 switching. In the L5 mode, the SLB used HTTP and HTML headers to make routing decisions. All routing decisions were made in real time (1 GBPs) by hardware logic in FPGAs while a Power PC provided a user interface, configuration management and housekeeping diagnostics. Communications between the hardware elements and the PowerPC CPU was via a PCI bus. I was responsible for the system design as well as for managing the product manufacturing process.

13. From January 2000 to March 2001, I was Director of Hardware for Webstacks, Inc. (now Extreme Networks, Inc.). Extreme Networks and several venture founding partners commissioned Webstacks to build a Content Addressable Switch which would operate as a L2-5 server load balancer. This switch included content routing, load balancing, firewall and security capabilities utilizing Secure Sockets Layers (SSL). I was hired to be the system architect. My responsibilities included hiring the hardware and software engineers that would implement the server load balancer as well as set up the manufacturing process. I hired 13 ASIC, FPGA and software engineers into my department and managed them for the

duration of the project. Within one year, our team had developed the product and met Extreme Networks' acquisition criteria. Extreme Networks purchased Webstacks for \$68MM cash and stock.

14. From February 1997 to December 1999, I was a consultant for Philips Semiconductor. I developed Philips Semiconductors' portion of a certification environment used by Philips Semiconductor and Microsoft to verify operation of Philips' Poseidon handheld chipset running Windows CE. This included board development, C coding, FPGA coding and simulation as well as porting to Windows CE. Later, I supported the development of Philips' version of the MIPS 4000 with communications and storage devices to emulate a system on a chip. I also designed ASIC logic and board level hardware for a 56Kbs software modem while at Philips.

15. From September 1994 to January 1997, I was President and Founder of Minden Group, Inc. The Minden Group developed and sold several types of memory adapters and a video conferencing system. The adapters were designed to expand the memory capacity of Personal Computers while saving the customer money. Both the adapters and the video conferencing system were sold through retail stores throughout the United States and Canada. Over 250,000 memory adapters were sold at stores such as Fry's, CompUSA, Computer City and Future Shop. After developing these products, I organized and managed a team of 10 to 15

employees focused on sales, marketing, finance and production. Manufacturing of the primary components was performed through contract manufacturing in Taiwan with final assembly and distribution in the United States. Distribution included order fulfillment, contracted sales channel distribution networks and end-point retail management.

16. From August 1992 to September 1994, I was Senior Hardware Manager of RAID Product Development for MTI, Inc. In this capacity, I was responsible for architectural development of the next generation RAID 1-5 systems and for managing the product development group. I introduced several design approaches that resulted in lower product costs as well as methods for interfacing with the company's other hierarchical storage components.

17. From April 1989 to August 1992, I was Founder and President of Spectrum Analysis, Inc., a consulting services company. I founded and operated Spectrum Analysis, Inc. with three other partners. Spectrum Analysis, Inc. specialized in electrical circuit design with emphasis on FPGA and ASIC emulation and complex PCB level designs.

18. From September 1988 to March 1989, I was Applications Manager for Quickturn Systems, Inc. Quickturn developed an ASIC emulator using Xilinx FPGAs. I was hired as Quickturn's first applications engineer, and was initially

responsible for developing the training and demonstration systems for Quickturn's initial customers. Later, as the company matured, I managed the customer support team. Because this ASIC emulator worked at a fraction of clock speed, I developed a methodology of slowing board designs to a fraction of the original clock speed, (on the order of 100s of KHz) while maintaining clock and memory (DRAM) integrity.

19. From July 1987 to August 1988, I was an Applications Engineer for Telestream Corporation and was responsible for demonstration system development, training and sales support for a software based communications protocol converter product. From March 1985 to July 1987 I was a design engineer for StrataCom Inc., an early voice and data network equipment manufacture. StrataCom developed the IPX, a packet over leased T1 transmission and management system. I was responsible for the voice compression card as well as designing a way to include data over a primarily voice solution. The hardware I developed eventually evolved into the Frame Relay protocol. Cisco bought StrataCom in 1996.

20. From June 1983 to March 1985, I was a Design Engineer for ROLM Corporation. I was responsible for the development of a multiplex/demultiplex hardware board used in the ROLM BUS 295 telephone switch. A novel elastic buffer was used to automatically determine the length of an inter-cabinet data bus,

whose length varied based on the customer's requirements. This eliminated the need for manufacturing technicians to manually adjust clock skew.

21. A more detailed account of my work experience and other qualifications is listed in my Curriculum Vitae filed as Exhibit 2026 in this matter.

III. BASES OF OPINIONS

22. In the course of conducting my analysis and forming my opinions, I have reviewed the exhibits submitted to date in this IPR, as well as the Petition, Preliminary Response, and Institution Decision.

IV. APPLICABLE LEGAL STANDARDS

A. Ordinary Skill in the Art

23. My opinions in this declaration are based on the understandings of a person of ordinary skill in the art, which I understand is sometimes referred to as an “ordinary artisan” or by the acronym “POSITA” or “POSA” (person of ordinary skill in the art), as of the time of the invention. For purposes of this Declaration, I have been instructed to use April 7, 2006 as the time of invention. I understand that the person of ordinary skill in the art is a hypothetical person who is presumed to have known the relevant art at the time of the invention. By “relevant,” I mean relevant to the challenged claims of the '111 Patent.

24. I understand that, in assessing the level of skill of a person of ordinary skill in the art, one should consider the type of problems encountered in the art, the prior solutions to those problems found in the prior art references, the rapidity with which innovations are made, the sophistication of the technology, the level of education of active workers in the field, and my own experience working with those of skill in the art at the time of the invention.

25. I was at the time of invention, and am currently, one of more than ordinary skill in the art through my education and research experience. As of the date of the invention, I am very familiar with the types of problems encountered in computer networking , the types of prior art solutions described in prior art references, and the rapidity at which innovations are made. Indeed, I am very familiar with people having this level of skill in the area of computer networking and am very familiar with commonly used technologies as Ethernet and the internet protocol / transmission control protocol (TCP/IP) suite. At the time of the invention, and since that time, I have been involved in the design and development of computer networks and in that capacity have worked with and managed teams of numerous individuals that would qualify as a POSITA.

B. Claim Construction

26. I understand that claims of the patent-at-issue in this IPR are generally interpreted according to their ordinary and customary meaning taking into consideration the so-called “intrinsic evidence” of the patent consisting of (1) the claim language; (2) the specification; and (3) the prosecution history.

27. I understand that claim terms may be explicitly defined in the patent specification, or they may be implicitly defined through consistent usage in the specification. I also understand that the scope of claim terms may be limited by statements in the specification or prosecution history where the applicant clearly disavows or disclaims subject matter.

C. Obviousness (35 U.S.C. § 103)

28. I have been informed that a patent may be invalid if the claimed invention would have been obvious at the time the invention was made to a person having ordinary skill in the art. I have been informed that the following factors must be evaluated to determine whether Petitioner has met its burden of proof that a claimed invention is obvious:

1. The scope and content of the prior art relied upon by Petitioner;
2. The difference or differences, if any, between each claim of the patent and the prior art;

3. The level of ordinary skill in the art at the time the invention of the patent was made; and
4. Any objective indicia of non-obviousness, including, for example:
 - (1) commercial success of an embodiment; (2) a long-felt need; (3) skepticism; (4) failure by others to find the solution provided by the claimed invention; (5) copying by others of the subject matter of the claim invention; (6) unexpected results of the claimed invention; (7) acceptance of others and industry praise; and (8) licensing of the patents.

29. I have been informed that a claim is not proved obvious merely by demonstrating that each of the elements was independently known in the prior art. I have been informed that many, if not all, inventions rely on building blocks already known, and claimed inventions almost of necessity will likely be combinations of what is already known.

30. I have been informed that it is important in the obviousness inquiry to identify whether a reason existed at the time of the invention that would have motivated a person of ordinary skill in the art in the relevant field to combine or modify the prior art references in the manner proposed by the Petitioner to arrive at the claimed invention. Put another way, a finding of obviousness should be

supported by an apparent reason to combine or modify the prior art references as proposed by the Petitioner.

31. I have been informed that it is important in the obviousness inquiry that it is understood how the combination of references is supposed to work. An explanation of the operation of the combined references is often a prerequisite to showing that a person of ordinary skill in the art would have been motivated to make the proposed combination and would have had a reasonable expectation of success.

32. I have been informed that a finding of obviousness “requires the additional showing that a person of ordinary skill at the time of the invention would have selected and combined those prior art elements in the normal course of research and development to yield the claimed invention.” *Unigene Labs., Inc. v. Apotex, Inc.*, 655 F.3d 1352, 1360 (Fed. Cir. 2011). In addition, I have been informed that a person of ordinary skill in the art should have had a reasonable expectation of success in combining the references as proposed.

33. In assessing obviousness, I have been instructed to consider both the ordinary creativity and common sense of the person of ordinary skill in the art. I also understand that it is impermissible for common sense to be applied to fill gaps in prior art that fails to teach or suggest a limitation of the claim. I also understand that the obviousness inquiry should guard against hindsight bias or hindsight

reconstruction where after-the-fact reasoning is applied to combine prior art elements using the claimed invention as a template, without establishing that, as of the date of the invention, there exists a motivation to combine or apparent reason to combine the prior art as proposed.

34. In assessing obviousness, I have been instructed that, to qualify as proper prior art for an obviousness analysis, a reference must qualify as analogous art. I have been informed that a reference qualifies as analogous art with respect to the claims if it is either: (1) from the same field or endeavor as the patent; or (2) the reference is reasonably pertinent to the particular problem addressed by the invention. I have also been informed that for a reference to be reasonably pertinent, it must logically have commended itself during the ordinary course of development to an inventor's attention in considering his problem.

35. In assessing obviousness, I have also been instructed that, as noted above, secondary considerations can often be probative evidence of non-obviousness and can serve to avoid hindsight bias. I have been instructed that secondary considerations can be sufficient to overcome a prima facie case of obviousness. I have been instructed that Patent Owner can show non-obviousness if: (1) the industry praised their work; (2) their design was copied by others; (3) their design

achieved a high level of commercial success; or (4) their invention had unexpected beneficial results, for example.

V. OVERVIEW OF THE '111 PATENT AND THE STATE OF THE ART

A. Technical Background of the '111 Patent

36. The '111 Patent is titled “Method and System for Deep Packet Inspection in Software Defined Networks.” Ex. 1001, abstract. Deep Packet Inspection (DPI) is a technique for examining network communications that can be used to extract data patterns from a data communication channel. Ex. 1001, 1:21-25. The extracted data patterns are useful for a variety of purposes, including network security and data analytics. *Id.*

37. A software defined network (SDN) is a networking architecture that provides for centralized management of the nodes in a network, as opposed to the distributed architecture utilized by conventional networks. Ex. 1001, 1:30-38. For example, a SDN may utilize a controller to manage network nodes such as vSwitches. *Id.* SDN-based architectures typically decouple the data forwarding (*e.g.* data plane) from control decisions (*e.g.* control plane), such as routing, resources, and other management functionalities. Ex. 1001, 1:39-49. The decoupling may allow the data plane and the control plane to operate on different hardware, in different runtime environments, and/or operate using different models. *Id.*

38. The '111 Patent proposes methods for DPI in a software defined network. Ex. 1001, 2:27-3:3. In one exemplary embodiment, a network system 100 as shown in Fig. 1 includes a controller 111 “configured to perform deep packet inspection on designated packets from designated flows or TCP sessions... the central controller 111 is further configured to instruct each of the network nodes 112 which of the packets and/or sessions should be directed to the controller 111 for packet inspections:”

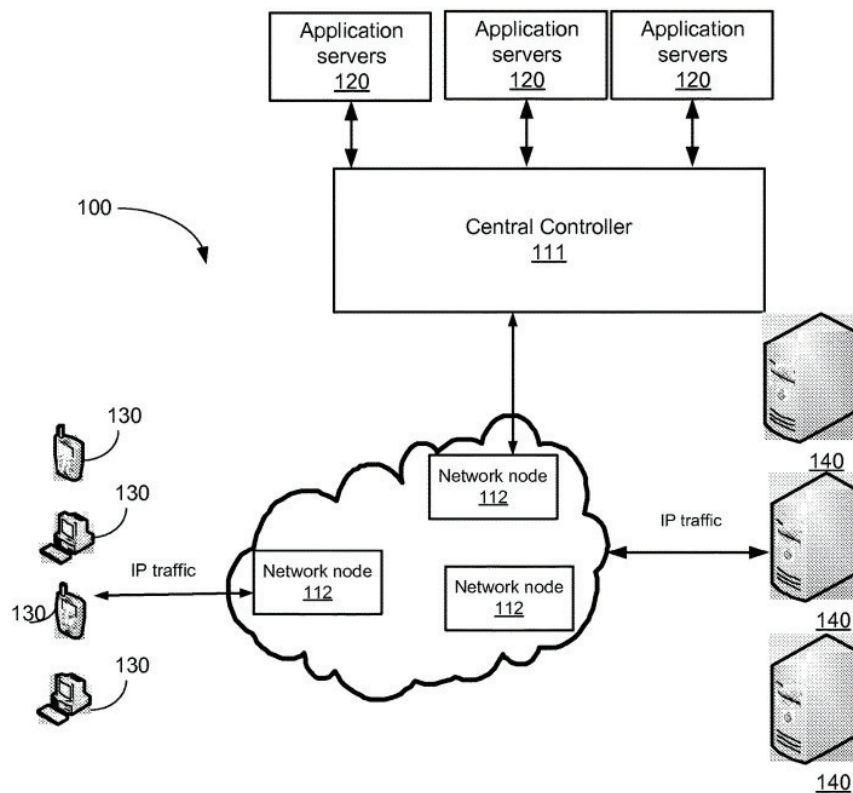


FIG. 1

Ex. 1001, 4:5-11, Fig. 1.

39. Controller 111 may provide instructions to the network nodes that configure the network nodes to analyze the headers of incoming packets for DPI. Ex. 1001, 4:23-37. For example, the network node may be instructed to either redirect the packet to controller 111 or send the packet to the destination server 140. *Id.* Alternatively, the controller 111 may instruct the network nodes to mirror a number of bytes within the packet to be sent to controller 111 to perform DPI. *Id.* The TCP packet's header may include fields such as TCP FLAG SYN, TCP FLAG ACK, TCP FLAG FIN, TCP FLAG RST, which are analyzed by the network node to execute the instructions sent by the controller, which may vary depending on the contents of the TCP packet's header field. Ex. 1001, 4:23-46.

B. Petitioner's Cited References

40. The Petition asserts the following two grounds:

Ground	Claims	Basis
#1	1-9, 12-24, and 27-31	35 U.S.C. § 103 (AIA) over Lin and Swenson
#2	1, 5-9, 12-24, and 27-30	35 U.S.C. § 103 (AIA) over Shieh and Swenson

Pet., i-iii.

1. Lin (Ex. 1005)

41. Lin (U.S. Patent No. 9,264,400) is titled "Software Defined Networking Pipe for Network Traffic Inspection" and describes a software defined networking (SDN) computer network including an SDN controller and an SDN switch. Ex.

1005, abstract. The SDN controller inserts flow rules in a flow table of the SDN switch to create an SDN pipe between a sender component and a security component. *Id.*

42. Lin utilizes an SDN Controller 610 (which may be an OpenFlow Controller) that controls the behavior of the SDN computer network 600 as shown in Fig. 6:

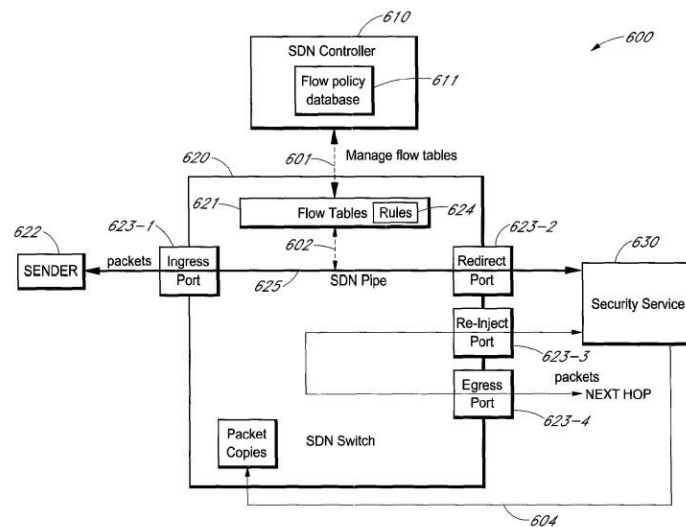


FIG. 6

Ex. 1005, 3:40-52, Fig. 6.

43. The SDN controller may include a flow policy database 611. *Id.* The flow policy database comprises flow policies that are enforced by the controller 610 on network traffic transmitted over the SDN computer network 600 and may specify security policies that govern transmission of packets on the network. *Id.* Flow

policies in database 611 are implemented in terms of flow rules 624 that are stored in flow tables 621 of the SDN switch 620. Ex. 1005, 4:19-31. “The SDN controller 610 may insert flow rules in the flow tables 621.” Ex. 1005, 6:1-3.

2. Shieh (Ex. 1006)

44. Shieh (U.S. Pat. Pub. No. 2013/0291088) is titled “Cooperative Network Security Inspection” and describes a network system including a security device and a network access device. Ex. 1006, abstract. The network access device receives a packet from a source node of the network and determines whether the packet should be routed to a destination node or forwarded to a security device. Ex. 1006, [0037]. The network access device makes this determination based on whether a bypass flag in a session table stored on the network access device is set to a predetermined value or not. *Id.* If the bypass flag is the predetermined value, the packet is forwarded to the security device. *Id.* Otherwise, the packet is routed to the destination node. *Id.* The bypass flag may be received from the security device, such as to indicate that the security device no longer wishes to receive further packets for security inspection. Ex. 1006, [0030].

3. Swenson (Ex. 1007)

45. Swenson (U.S. Pat. Pub. No. 2013/0322242) is titled “Real-Time Network Monitoring and Subscriber Identification with an On-Demand Appliance”

and describes a system and method for selectively monitoring traffic in a service provider network. Ex. 1007, abstract. A user device 110 accesses a network via a steering device 130. Ex. 1007, [0026]. A network controller 140 interfaces with steering device 130 to coordinate the monitoring and categorization of network traffic. *Id.* The network controller provides selective real-time network monitoring and subscriber identification and monitors large flows in the network, collects flow statistics, and selects flows to be optimized. Ex. 1007, [0039-0040]. The network controller may monitor and optimize flow based on historical traffic data for the average of the bandwidth per user at a particular cell tower to estimate the amount of bandwidth or degree of congestion for the new flow based on the historical record. Ex. 1007, [0029].

C. Level of Ordinary Skill

46. The Petition and Petitioner's expert assert:

[A] POSA would have had a working knowledge of hardware and software for packet-switched networking. A POSA also would have had a bachelor's degree in computer science, computer engineering, or an equivalent, and two years of professional experience. Further, it is my opinion that a person with additional education but less professional experience may still qualify as a POSA, and a person with additional professional experience but less education may still qualify as a POSA.

Ex. 1004, ¶48. I do not dispute Petitioner’s level of ordinary skill for purposes of this declaration.

47. My level of qualifications as described in Section II meet this level of ordinary skill at the time of invention of the ’111 Patent.

VI. CLAIM CONSTRUCTION

48. I understand that Petitioner has proposed the term “controller” to mean “an entity configured to perform deep packet inspection on packets” whereas Patent Owner has proposed “controller” to mean “an entity that is capable of controlling deep packet inspection,” which I have been informed is consistent with the construction of “controller” in a parallel District Court case. My opinions as set forth below are the same under either parties’ proposed construction of “controller.”

VII. NO COMBINATION OF THE CITED PRIOR ART REFERENCES TEACHES OR SUGGESTS EVERY LIMITATION OF THE CLAIMS

A. Lin and Swenson do not render obvious claim 1

1. Lin and Swenson do not disclose or render obvious the claimed controller under either Patent Owner’s or Petitioner’s proposed constructions.

49. Claim 1 requires a “controller.” I understand that Petitioner has construed the “controller” as “an entity configured to perform deep packet inspection on packets” while Patent Owner has construed the “controller” as “an entity that is capable of controlling deep packet inspection.” My opinion does not change based

on either construction. In either case, the “controller” must “control[] deep packet inspection.”

50. I understand that Dr. Bhattacharjee’s opinion is that “Lin also discloses that the SDN switch (i.e., the network node) is under the control of a “SDN controller” that is external to the SDN switch.” Ex. 1004, ¶78. His opinion is that “the combination of Lin and Swenson discloses a controller configured to perform DPI” because Lin discloses DPI and Swenson “Swenson teaches the use of a controller configured to perform DPI.” Ex. 1004, ¶79-80. In other words, Petitioner’s theory appears to be that either (i) a POSA would implement DPI as part of Lin’s controller, notwithstanding that Lin discloses a security service entity for DPI that is separate from the controller, or (ii) a POSA would have been motivated to combine Lin with Swenson, which Petitioner asserts discloses a controller that is capable of performing DPI.

51. I disagree. In my opinion, Lin does not disclose a controller (under either party’s construction) that is configured to perform (or capable of controlling) DPI. Swenson likewise does not disclose the claimed controller. In addition, a POSA would not have been motivated to combine Lin and Swenson to practice this limitation in the manner required by claim 1.

- a. **Lin does not disclose or render obvious a controller configured to perform or capable of controlling DPI.**

52. Lin does not disclose a controller that is configured to perform or capable of controlling deep packet inspection. Lin teaches that DPI is performed by security service 630, not controller 610. Ex. 1005, 3:11-24, 4:63-67 (“allowing the security service 630 to inspect packets”). As shown in the figures excerpted from Lin, the security service 630 is a separate entity from the controller 610:

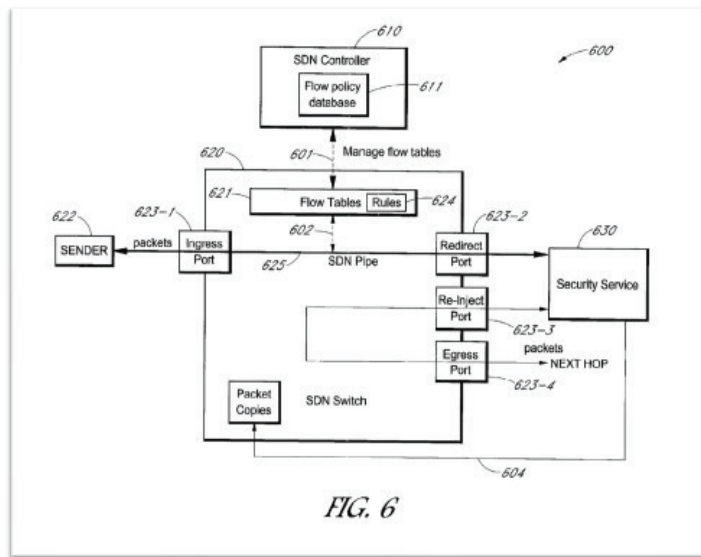


Figure 6 of Lin

53. The Petition and Dr. Bhattacharjee alternatively assert that a POSA would nevertheless “have understood that one of the limited number of design options would have been to implement the security service as part of a controller configured to perform DPI analysis on packets, and a POSA would have had a reasonable expectation that the controller would have been successful in performing DPI analysis.” Pet. 18-19; Ex. 1004, ¶79. As support, Dr. Bhattacharjee states that a

“POSA would have understood these disclosures in Lin to teach that the security service 630 can be implemented using the same hardware or software as the controller, and that the security service 630 can be connected to the SDN switch 620 in the same way as the controller.” *Id.*

54. I disagree that a POSA would modify Lin’s security service in this manner based on Lin’s disclosure. The mere fact that the security service may share certain common technical characteristics with the controller (i.e., its connection to the network switch and the hardware in which these two separate entities are implemented) cannot in and of itself suggest to a POSA that the function carried out by the security service should also be performed by the controller or that the controller should be configured to perform that function (i.e., deep packet inspection).

55. As discussed above, Lin teaches that the security service is a discrete network element separate from the controller. Additionally, Lin explicitly separates network security services (including DPI) from the controller (or the network administrators responsible for configuring the controller), stating: “[n]etwork security vendors provide network security services, such as firewall or deep packet inspection (DPI)” (Ex. 1005, 3:11-12). Lin teaches that “to provide network security services, packets of network traffic are intercepted for inspection [by] plac[ing] the

security service in the middle of the packet forwarding path.” *Id.* at 12-16. These disclosures in which packets are routed for inspection to an entity that is entirely discrete from the controller are consistent with teachings in the ’111 patent specification about the prior art, which states in the background section that the “straightforward approach of routing all traffic from network nodes to the central controller introduces some significant drawbacks, such as increased end-to-end traffic delays between the client and the server; overflowing the controller capability to perform other networking functions; and a single point of failure for the re-routed traffic.” ’111 patent, 2:1-5. Thus, it was known in the art before the ’111 patent that performing DPI at the controller presented material challenges and disadvantages. Indeed, a POSA at the time of invention would understand that security services were not normally combined with a controller because the security service requires security resources not normally found in a controller or that would be prohibitively expensive to implement as part of the controller. Thus, the ’111 patent—but not Lin—explicitly addresses the issues arising from performing DPI at the controller. On the other hand, Lin alone does not give a POSA any reason to implement the security service as part of the controller, nor a reasonable expectation of success in doing so.

b. Swenson does not disclose or render obvious a controller configured to perform or capable of controlling DPI.

56. Alternatively, the Petition and Dr. Bhattacharjee contend that a POSA modify Lin to arrive at the claimed controller because Swenson “teaches the use of a controller configured to perform DPI.” Pet. 19. Specifically, Petitioner points to Swenson’s “flow analyzer 312” that performs “deep flow inspection to determine if the flow is worth bandwidth monitoring and/or user detection” to assert that a “POSA would have known that a ‘flow’ is a series of packets” and “[a]s such, it would have been obvious to a POSA that Swenson’s reference to ‘deep flow inspection’ refers to performing DPI in one or more packets in a flow.” Pet. 19. I disagree that a POSA would combine Lin and Swenson as alleged.

57. A POSA would understand that Lin discloses deep *packet* inspection (DPI) performed at a dedicated security service, whereas Swenson discloses deep *flow* inspection (DFI) performed in the flow analyzer of a network controller. Thus, Lin and Swenson disclose *different* techniques, which a POSA would understand from the use of different terms of art in the two references.

58. Furthermore, a POSA would understand that deep *flow* inspection is not the same as deep *packet* inspection. **Deep Packet Inspection** (or “DPI”) is well-known technique used for computer network packet filtering that may examine a

packet payload (i.e., the part of the packet that contains content relevant to an application, such as an email message or audio/video content) and in some instances the packet header, as a packet is transmitted through an inspection point (e.g., a node, controller, or security device). Networks use DPI for a number of purposes that include protocol non-compliance, viruses, spam, intrusions, or other defined criteria applicable to the packet. On the basis of DPI, networks then decide whether particular data packets and possibly the flows associated with those packets should be routed to the destination to which they are addressed or some other destination, or alternatively, whether the packets should be dropped entirely. DPI is a thorough inspection process that can be utilized for a variety of purposes, which include inspecting and managing network traffic, enforcing network security (e.g., by blocking malware or illegal content). DPI devices are capable of looking at Layer 2 and beyond Layer 3 of the OSI model, and can even analyze up to Layer 7. DPI thus offers control of network traffic that goes beyond classification of packets based on header information and can be used in enterprise security to detect various attacks, including buffer overflow attacks or data leaks.

59. DPI also presents certain unique challenges. Because it is a thorough packet inspection technique, DPI requires significant processing power and may struggle with high-bandwidth networks or encrypted traffic. Further, DPI tools may

have limited visibility into the overall traffic patterns of networks in which less traffic flows through core switches (e.g., in modern distributed network environments). In such scenarios, less granular metadata analysis techniques may be deployed to overcome certain limitations of DPI by giving a more comprehensive view of the entire network, without being affected by limitations that inhere in DPI.

60. On the other hand, **Deep Flow Inspection** (DFI), is a less well-known technique that—in general—refers to a type of network monitoring focused on the flow of data within the network rather than the inspection of individual packets. Flows convey information that is relevant to different aspects of network monitoring than the information conveyed by individual packets. In particular, inspection of flows conveys information about the state of the network that is not conveyed by individual packets.

61. Flow data typically includes information about the IP addresses, ports, and protocol types involved in a communication session, as well as metrics like the number of bytes and packets transferred. DFI can be used for purposes like traffic analysis, network performance monitoring, and the detection of unusual or suspicious behavior patterns in network traffic.

62. For these reasons, a POSA would have known that DPI inspects the detailed contents of individual packets in a granular, thorough manner; and, in

contrast, DFI focuses on the broader patterns of network traffic flows by looking at aggregate data flow characteristics. A POSA would understand that the two techniques entail material differences, and that deep packet inspection entails unique technical capabilities and challenges that distinguish it from deep flow inspection.

63. This understanding—*i.e.*, that deep flow inspection is a materially different technique than deep packet inspection—is consistent with Swenson’s disclosures of a system geared toward bandwidth and user experience optimization through the use of statistic information relating to flows, not the inspection of packet contents. I excerpt examples of these disclosures in Swenson below:

- The system then determines whether to monitor the data flow from the origin server to the user device. If so determined, *the system collects statistic information of the data flow and stores the statistic information* to a flow record in a database. The system also maps the flow record to a subscriber of the service provider network by *analyzing the statistic information of the data flow and estimates bandwidth provided to the data flow by the service provider's network based on the analysis of the statistic information of the data flow*. Ex. 1007, abstract.

- The disclosure generally relates to improving user experience on a network, and more specifically, to monitoring bandwidth consumption of the many devices connected to a given node in the network. Ex. 1007, [0003].
- Embodiments disclosed include a network controller system for real-time data gathering on the state of existing data network traffic flows and mapping flow data to respective users in the network to predict available bandwidth and level of congestion. By gathering a history of flow statistics in the network, the network controller system establishes a relationship between base stations (or other network segments) and their capability to deliver the amount of data typically required by a particular user of the network. The very recent history of network flows can be used to predict the near future congestions in a substantially real-time fashion. Furthermore, the history of flow statistics can be used to build a long-term map of user behavior on the network, which can more effectively predict on demand data delivery requirements for the collection of users utilizing a given network access point in a consistent manner. The network

controller keeps a flow state database, which groups flow data in a number of ways, such as on per station/cell tower, per subscriber, per time-of-day, or per geography area basis. As new flows are presented to the system for inspection, database can be queried to estimate the network congestion level for new flows to determine whether existing, new or future flows require optimizations in order to maintain the desired level of user satisfaction. Ex. 1007, [0018].

64. Thus, Swenson itself teaches that deep *flow* inspection is not the same as deep *packet* inspection. In my opinion, DPI would have little use for the purposes of Swenson's traffic analysis due to the higher overhead it would impose upon the Swenson system as a whole.

65. As a result of these differences, it is my opinion that a POSA would not combine a controller performing deep flow inspection (e.g., Swenson's controller) with a network in which deep packet inspection is performed in a dedicated security service that is discrete from the controller (e.g., Lin's network) to practice the claimed invention. For similar reasons, a POSA would understand that Swenson's controller is not configured to perform (or capable of controlling) deep packet inspection.

66. Nothing in the Petition satisfactorily addresses the distinctions between deep packet inspection and deep flow inspection. While the Petition asserts that flows are “series of packets having a specific signature,” that does not address the material differences between these two techniques such that a POSA would glean from Swenson a controller that is configured to perform deep packet inspection.

67. Finally, I want to address Petitioner’s assertion that a POSA would understand Swenson’s discussion of a controller that “inspects the message” including a HTTP request header and payload portion received from a steering device “to determine whether to monitor the flow or optimize the video” to refer to “DPI by a DPI-capable controller, as DPI refers to monitoring the payload of a packet” (i.e., Swenson’s controller is configured to perform (or capable of controlling) DPI. Pet., 19.

68. Swenson’s disclosure of the controller analyzing a message does not require analyzing the message payload. Swenson does not disclose analyzing the message payload; rather, Swenson says that the message is inspected without stating that the message payload is also analyzed. Swenson’s disclosure, which is geared toward flow monitoring rather than individual packet monitoring, does not suggest that the controller analyzes an individual packet payload and even if it did that such an analysis would constitute DPI.

69. Furthermore, Swenson discloses that the steering device 130 “sends an ICAP request message 516 comprising [a] HTTP GET request header and a portion of the [HTTP] response payload to the network controller 140, which inspects the message to determine whether to monitor the flow or optimize the video.” Ex. 1007 [0065]. In the context of a HTTP GET request and response, as disclosed in Swenson, the message does not constitute a single packet but a *flow* of packets. The reference to a HTTP request and response “header” and “payload” are thus not references to the header and payload of a packet. A POSA would know that despite common parlance references to “HTTP packets,” HTTP does not pertain to individual packets, whereas packet transmission is governed by the TCP protocol. HTTP utilizes TCP to exchange data such that information may be used by an end user (e.g., a video or audio stream). Even assuming that Swenson’s controller were analyzing the header and payload of an HTTP request/response, a POSA would understand that this analysis is not at the packet but at the flow level and thus does not constitute deep packet inspection.

c. A POSITA would not have combined Swenson’s controller with Lin’s or adapted such a combination to practice the claimed controller in Lin’s system in any event.

70. Finally, a POSA in my opinion would not have combined Swenson’s controller with Lin’s system, even if Swenson disclosed (it does not) a controller that

satisfied either Petitioner's or Patent Owner's proposed construction (i.e., an entity configured to perform or capable of controlling DPI).

71. The Petition asserts that Lin and Swenson "use a central controller to provide instructions and packet-applicable criterion to network nodes to determine which packets should be redirected and which packets can be sent directly to a destination node." Pet. 20. A POSA however would understand that, even assuming Swenson discloses a packet-applicable criterion, it is directed to a much narrower function than Lin (bandwidth monitoring) and deploys an altogether different technique (deep flow inspection) in the controller that is much less taxing on the controller than the deep packet inspection, which is performed by a altogether different device in Lin (the security service). Because deep packet inspection in Lin is much more resource-intensive than the deep flow inspection taught in Swenson, a POSA would not have been motivated to adapt Lin in accordance with Swenson to execute deep packet inspection (which Lin expressly teaches is conducted by specialized "security vendors") in the SDN controller responsible for managing the SDN control plane.

72. The Petition further asserts that "[l]ike Lin, Swenson teaches that only certain packets are redirected to the controller in order to be efficient in analyzing packets of interest." Pet. 21. I disagree that this accurately captures Swenson's

disclosure. Swenson teaches forwarding an “HTTP request and a portion of the HTTP response to the network controller 140 over the ICAP client interface 404.” Ex. 1007, [0060]. This disclosure however refers to directing HTTP messages—which can consist of numerous packets—to the controller for analysis of that message, rather than deep inspection of any individual packet. A POSA would not understand this disclosure to teach routing packets for deep packet inspection. Thus, a POSA would not conclude from Swenson’s disclosure of deep flow inspection at the controller that it is also possible to conduct DPI at the controller.

73. I also disagree with Petitioner’s assertions that a POSITA would have understood that Swenson, like Lin, could involve analyzing packets for a security function (Pet., 21) or that Swenson discloses “send[ing] packets to the controller for DPI” (Pet., 24). Swenson states that in “other embodiments, the network controller 140 may include additional, fewer, or different components for various applications ... [for example, conventional] components such as network interfaces, security functions, failover servers, management and network operations consoles,” which “are *not shown so as to not obscure the details of the system architecture.*” Ex. 1007, [0039] (emphasis added). Far from disclosing specific techniques for inspecting packets or security such as DPI, Swenson expressly omits technical details pertaining to security (such as DPI or how DPI might be conducted in the

context of a SDN) from its disclosure because they are *irrelevant* to its system and would only serve to *obscure* the details of its system.

74. A POSA would understand from Swenson’s disclosure that in certain embodiments “the bandwidth attained by a single flow may be sufficient to determine the capacity of the network segment(s) the flow traverses” such that a “fairly small deployment of network controller(s)” is sufficient for obtaining “an accurate detection of key network congestion points” without a “need to monitor every flow going through a network segment to detect congestions” that Swenson is focused on high-level statistical analysis of network traffic patterns—as opposed to granular, packet-based analysis such as DPI. Ex. 1007, [0021]. Swenson focuses on network bandwidth and traffic congestion and does not pertain to network security, let alone resource-intensive techniques such as DPI or the specific manner in which such techniques might be executed or optimized. Nothing in Swenson constitutes a disclosure of a controller that is capable of DPI or that would warrant incorporation of a function as complex and taxing as DPI into another system, let alone a system such as Lin, which expressly teaches the use of a dedicated security apparatus, wholly divorced from the SDN controller, to perform DPI.

75. I also disagree with Petitioner’s assertion that “Lin and Swenson each teach that only certain packets may need to be redirected for further processing

before being sent to the destination node.” Pet., 23. Lin expressly teaches performing DPI at a dedicated security service that is divorced from the controller—which is consistent with the ’111 patent’s disclosure discussed *supra* that performing DPI at the controller was a known drawback in the prior art prior. Swenson is not focused on DPI at all, and expressly *omits* controller-based security functions like DPI. Nothing in these disclosures suggest to a POSA to either to combine Lin and Swenson or to otherwise practice the claimed controller (i.e., a controller configured to perform or capable of controlling DPI).

76. In any event, Lin does not contain any teaching that its security service—which is the device used to perform DPI—could or should be implemented in the controller. Performing DPI at the controller would thus be duplicative of a function already performed in Lin. Accordingly, the art does not inform a POSA to adapt Lin to provide a controller configured to perform (or capable of controlling) the resource-intensive DPI function.

- d. Lin does not disclose sending the packet to the second entity “responsive to the packet not satisfying the criterion” or sending the packet to an entity other than the second entity “responsive to the packet satisfying the criterion.”**

77. Claim 1 requires (i) “receiving, by the network node from the controller, the instruction and the criterion”; and then either (ii) “responsive to the

packet not satisfying the criterion, sending, by the network node over the packet network, the packet to the second entity”; or (iii) “responsive to the packet satisfying the criterion, sending the packet, by the network node over the packet network, to an entity that is included in the instruction and is other than the second entity.” Thus, Claim 1 requires a routing decision—*i.e.*, whether to send the packet to the second entity to which it is addressed or to a different entity—based on whether the packet satisfies the criterion received “by the network node from the controller.”

78. In my opinion, the Petition incorrectly relies on two separate criteria to satisfy this limitation, whereas Claim 1 requires a single criterion be used to make the claimed routing decision (*i.e.*, based on whether that criterion is satisfied or not satisfied). Using two separate criteria as alleged cannot meet using a single criterion to make the decision as claimed.

79. First, the Petition identifies the condition used by Lin’s “bypass flow rules” as the criterion responsive to which Lin’s SDN switch sends the packet to the second entity. Pet. 31. The Petition cites Lin’s Table 3 (reproduced as annotated by Petitioner below), in which “the bottom two rows are bypass flow rules for all packets:”

IN_PORT	...	IP src	TCP src port	TCP dst port	...	Action	Count
Ingress_port_ID	*	*	*	80	*	Redirect port	10
Redirect_port_ID	*	*	80	*	*	Ingress port	10
Ingress_port_ID	*	*	*	*	*	Egress port	130
Egress_port_ID	*	*	*	*	*	Ingress port	130

Table 3 of Lin (Annotated)

80. In my opinion, a POSA would understand from Lin’s disclosure that the bypass flow rules are set such that “packets bypass the SDN pipe, and are accordingly not inspected by the security service 630” if they do not indicate a TCP source or destination port.

81. While the Petition asserts that this same criterion is used in connection with “sending the packet, by the network node over the packet network, to an entity that is included in the instruction and is other than the second entity” (*see* Pet. 32-33), I disagree. In the same example given in Table 3, Lin makes clear that a different criterion is used in connection with sending packets to its security service. As Lin explains, “the top two rows are redirect flow rules for redirecting HTTP packets to the security service 630 for inspection,” and the criterion employed in connection with the “redirect flow rules” is the packet’s specific TCP source or

destination address, in this example (shown in Petitioner’s annotated Table 3), port 80:

TABLE 3							
IN_PORT	...	IP src	TCP src port	TCP dst port	...	Action	Count
Ingress_port_ID	*	*	*	80	*	Redirect port	10
Redirect_port_ID	*	*	80	*	*	Ingress port	10
Ingress_port_ID	*	*	*	*	*	Egress port	130
Egress_port_ID	*	*	*	*	*	Ingress port	130

Table 3 of Lin (Annotated)

82. Lin does not teach how a packet is sent if either the bypass flow rule and redirect flow rule conditions are both not satisfied. Lin’s rule tables only govern how a packet is sent if a particular condition is satisfied, as is evident from Lin’s discussion of the importance of rule priority in its table. Ex. 1004, 8:1-17. That is, Lin teaches that conditions are executed only if they are stated explicitly and does not teach actions to be taken on a packet if conditions are not satisfied or not explicitly stated.

83. A POSA would immediately understand that Lin uses two different criteria: one criterion for sending a packet to the security service (which Petitioner identifies as the claimed “entity that is other than the second entity”) and a separate criterion for sending the packet to the second entity. Lin’s conditions are thus

different than the claimed “packet-applicable criterion” (and the step of sending the packet to different entities in response to satisfying or not satisfying that criterion) in claim 1.

B. Shieh and Swenson do not render obvious claim 1

1. Shieh does not disclose the claimed “packet-applicable criterion” responsive to which a packet is sent either to a second entity or to another entity that is different from the second entity.

84. I understand that the Board’s Institution Decision previously found that Swenson does not disclose a “packet-applicable criterion.” Paper 8, 19. I agree.

85. I understand that the Petition asserted that Shieh’s “the identification of ‘TCP FIN or TCP RST packets’” used to “determine which packets should be forwarded to the security device pursuant to the filtering rules” are packet-applicable criterion. Pet., 62-63. In my opinion, these are not packet-applicable criterion for several reasons, including those identified by the Board.

86. First, “Shieh does not state whether these criterion [sic] are sent from a controller to the network access devices.” Pet., 62. Shieh describes the TCP FIN or TCP TST packet as being sent by the network source node and received by the network switch as part of the “packet stream” (Ex. 1006, 36). Even if “a POSA would have understood that the packet-applicable criterion would be send from a controller to one or more network access devices” as asserted by the Petition and Dr.

Bhattacharjee, Shieh does not inform a POSA how to send a TCP FIN or TCP TST packet from the controller to the network switch.

87. Second, the alleged packet-applicable criterion (TCP FIN or TCP RST) in Shieh fails to satisfy the remaining elements of claim 1, which requires “checking...if the packet satisfies the criterion” and then “sending...the packet to the second entity” or “sending...the packet to an entity that is included in the instruction and is other than the second entity” based on whether the packet satisfies the criterion. Claim 1 states that the packet is sent to different destinations based on whether or not the criterion is satisfied. On the other hand, Shieh discloses that packet destinations are determined by the bypass flag stored in the session table of the network switch, which the Petition has not alleged to be a packet-applicable criterion in element 1.1. Ex. 1007, 37; Paper 1, 62-63.

88. Shieh discloses that the bypass flag used to determine destination addresses is separate from the TCP FIN or TCP RST packet that the Petition alleges is the packet-applicable criterion. *Compare* Ex. 1006, [0037] (“[D]evice 272 may look up in its session table such as the one as shown in FIG. 5 to determine whether a bypass flag has been set to a predetermined value. If the bypass flag matches the predetermined value, the packet is forwarded to the security device 273 via path 282; otherwise, the packet is routed to destination node 274.”) with Ex. 1006, [0036]

(“During the bypass phase, the I/O function may notify the security-processing function if there are special events in the packet stream. These events could be receipt of TCP FIN or TCP RST packets...”). Shieh explicitly teaches that (i) the bypass flag is stored in the session table of the network switch (Ex. 1007, 37 (“device 272 may look up in its session table such as the one as shown in FIG. 5 to determine whether a bypass flag has been set to a predetermined value. If the bypass flag matches the predetermined value, the packet is forwarded to security device 273 via path 282; otherwise, the packet is routed to destination node 274”)); and (ii) the TCP FIN and TCP RST are packets received by the network switch from the source node. Ex. 1006, 31, 36 (“[W]hen a security-processing function (e.g., processing node 211A) receives the packets, it does the security inspection and security policy enforcement...During the bypass phase, the I/O function may notify the security processing function if there are special events in the packet stream. These events could be receipt of TCP FIN or TCP RST packets...”). Shieh is thus explicit that the bypass flag is different from the TCP FIN or TCP RST packets.

89. For these reasons, I agree with the Institution Decision’s determination that “it is unclear how identification of ‘TCP FIN or TCP RST packets’ corresponds to the teachings surrounding filtering rules in Shieh.” Paper 8, 19. I agree that “Shieh does not appear to teach what happens to a packet in a special event such as a TCP

FIN or TCP RST packet is identified” because “Shieh discloses that packet destinations are determined by the bypass flag—which is separate from the TCP FIN or TCP RST packet.” *Id.* In my opinion, the Petition incorrectly conflates TCP FIN and TCP RST packets, which Shieh discloses are “special events in the packet stream” that produce notifications to the security-processing function (Ex. 1006, 29), with the bypass flag stored in the session table of the network switch that flags packets to be sent to the security device for inspection (Ex. 1006, 30, 37).

C. Claim 3: The prior art does not disclose sending the packet to an entity that is other than the second entity *and* to the controller

90. Claim 3 depends from claim 1 and further requires “wherein the instruction is a ‘probe’, a ‘mirror’, or a ‘terminate’ instruction, and upon receiving by the network node the ‘mirror’ instruction and responsive to the packet satisfying the criterion, the method further comprising sending the packet, by the network node, to the second entity and to the controller.”

91. I have been informed that, if the claim language, as a matter of logic and grammar, specifies a series of steps, then the claim must be performed in that order of steps. In my opinion, a POSA would understand that the logic and grammar of claim 1 requires that claim to be performed in the stated order of steps.

92. Accordingly, it is my understanding that claim 1 requires a step of “sending the packet...to an entity that is included in the instruction and is other than

the second entity” and claim 3 then further requires the step of “sending the packet...to the second entity and to the controller.”

93. In my opinion, the Petition has not shown that Lin performs both claimed steps. The Petition states that Lin sends the packet to an entity that is other than the second entity on the basis of Lin’s disclosure that the packet is sent to the security device: “Lin states that, if a packet satisfies this criterion (i.e., the packet indicates that port 80 is the source or destination port), then the SDN switch (which corresponds to the claimed network node) sends the packet over the packet network to the security device (which corresponds to the claimed entity that is included in the instruction and is other than the second entity).” Pet., 32. But the Petition also asserts that Lin sends the packet to the controller based on the same disclosure of sending the packet to the security device: “Lin teaches that the mirror instruction instructs that ‘the SDN switch 620 may be configured to copy packets that are redirected to the security service 630 for inspection’ and to ‘forward the copied packets to their destinations.’ As discussed above for Element [1.6], it would have been obvious to implement the security service 630 in Lin as part of the controller; thus, the original packet would be sent to the controller by the SDN switch and a copy of the packet is sent to the destination node.” Pet., 35-36 (citations omitted).

94. At most, the Petition alleges that Lin performs one step of sending a packet once to a controller (I disagree with this assertion for the reasons stated above), but even if so, this single step does not meet two separate steps that separately send the packet (1) to an entity other than the second entity *and* (2) to the controller as required by claims 1 and 3.

D. Claims 4 and 5: The prior art does not disclose sending the packet to an entity that is other than the second entity *and* to the controller

95. In my opinion, claims 4 and 5 are not obvious in view of Lin for the same reasons claim 3 is not obvious in view of Lin. The Petition has identified only a single step of sending a packet to a controller in Lin, whereas, as discussed above, claim 4 requires both (1) “sending the packet...to an entity that is included in the instruction and is other than the second entity” and (2) “sending the packet...to the controller.” Similarly, the single step of sending a packet to a controller in Lin does not satisfy both (1) “sending the packet...to an entity that is included in the instruction and is other than the second entity” and (2) “sending the packet...to the controller” in claim 5.

E. Claim 6: The prior art does not disclose storing *the packet* by the controller in a memory

96. Claim 6 requires “storing the received packet or a portion thereof, by the controller, in a memory.”

97. I understand that Dr. Bhattacharjee’s opinion relies on Swenson’s “flow cache memory 322 that is part of the network controller 140.” Ex. 1004, ¶134. Specifically, “[t]he flow cache entry keeps track of the flow and its associated bandwidth.” *Id.*

98. In my opinion, Swenson’s flow cache memory 322 is at most a disclosure storing flow in a memory and does not suggest storing a packet in a memory.

99. As I explained above, a POSA would understand that a packet is an individual, granular unit of data whereas a flow is an aggregate of the data traffic and not granularized to any individual packet. A POSA would understand that deep packet inspection refers to examining a packet payload whereas deep flow inspection refers to monitoring the flow of data. Therefore, a POSA would understand that Swenson’s flow cache memory 322 is directed to caching *flow* data—not individual packets or portions of packets.

100. My opinion is confirmed by Swenson paragraph 73. The Petition and Dr. Bhattacharjee cites Swenson paragraph 73 as allegedly describing storing a portion of a packet: “In discussing Figure 7, Swenson states that the flow cache entry stores at least a portion of the received packet, including at least the IP address of the received packet. *Id.*, ¶[0073] (discussing Figure 7).” Pet., 40. However,

Swenson paragraph 73 in full explicitly states that the flow cache stores “flow cache entries indexed by a hash” and does not state that the flow cache caches packets:

FIG. 7 is a block diagram illustrating one embodiment of an example of internal components of the flow cache. The flow cache map 700 comprises a plurality of flow cache entries, such as flow cache entries 710 and 712 indexed by a hash. Not shown in the example diagram is a possible linked list behind each flow cache entry which allows chaining of flow cache entries for a given hash index. The hash into the flow cache may be based on source IP address, MAC address, subscriber ID, or other identifier indicative of a given subscriber, group of subscribers or subscriber's device.

Ex. 1007, [0073].

101. Furthermore, Swenson paragraph 73 does not state that an IP address is stored in the flow cache memory—paragraph 73 actually states that the “source IP address, MAC address, subscriber ID, or other identifier” are a basis for a hash index to retrieve the flow cache entries stored in the cache. Swenson describes the flow cache memory as storing a list of flows associated with subscriber IDs and flow statistics such as the number of bytes in either direction. Ex. 1007, [0076]. Swenson Paragraph 84 then states that the MAC address or TCP source port is the address or source port of the flow—not the payload of an individual packet.

102. Based on these disclosures, a POSA would understand that Swenson’s flow cache stores flow data and not packets or portions of a packet. Swenson does not disclose that any packet or portion of a packet is stored in the flow cache. Rather, the flow cache stores addresses of flow sources—not packets.

F. Claim 7: The prior art does not disclose sending “a portion of the packet” as claimed

103. Claim 7 requires “responsive to the packet satisfying the criterion and to the instruction, *sending a portion of the packet*, by the network node, to the controller.”

104. I understand that Dr. Bhattacharjee’s opinion is that “Swenson teaches that the network node sends a portion of a packet to the controller in response to the packet satisfying a criterion” because it “forwards the HTTP request and a portion of the HTTP response to the network controller 140.” Ex. 1004, ¶162. **I disagree that a portion of a HTTP response is a portion of a packet as in claim 7.**

105. A POSA would understand that a packet is an individual, granular unit of data. A POSA would further understand that an HTTP request or response is not a single packet but rather many packets. In my opinion, a POSA would understand that a portion of an HTTP request or response would still comprise multiple packets and would not understand a portion of an HTTP request or response to be a portion of a single packet.

106. This is consistent with Swenson itself, which describes the “portion” as a “portion[] of the network traffic”:

The steering device interface 316 interacts with an external routing appliance, such as the steering device 130 to divert **portions of the network traffic (e.g., large object network flows)...**

Ex. 1007, [0045].

107. When Swenson refers to diverting portions of HTTP requests/responses (i.e., network traffic), a POSA would understand that Swenson is referring to diverting network flow and is not referring to dividing a single packet into portions or diverting portions of packets.

G. Claim 16: The prior art does not disclose “the packet comprises distinct header and payload fields, the header comprises one or more flag bits, and wherein the packet-applicable criterion is that one or more of the flag bits is set”

108. Claim 16 requires “the packet comprises distinct header and payload fields, the header comprises one or more flag bits, and wherein the packet-applicable criterion is that one or more of the flag bits is set.”

109. I understand that Dr. Bhattacharjee’s opinion is that “it would have been obvious to a POSA that the flow rules of Lin could have used a determination of whether or not a flag bit was set as one of Lin’s ‘conditions’ that serve as packet applicable criterion.” Ex. 1004, ¶162. I note however that Lin does not teach a flag

bit, and certainly does not teach the use of a flag bit as a condition. Lin teaches that its conditions are indicated in a flow table (*see* Ex. 1005 5:8-24). Lin does not inform a POSA how to incorporate a flag bit in the flow table.

110. Similarly, I understand that Dr. Bhattacharjee’s opinion is that “a POSA would have found it obvious that the TCP FIN flag bit being set or the TCP RST flag bit being set [in Shieh] could function as the packet applicable criterion recited in Claim 1.” Ex. 1004, ¶280. Like for Lin, Shieh does not teach the TCP FIN flag bit being set or the TCP RST flag bit being set as a criterion. I agree with the Institution Decision’s determination that “it is unclear how identification of ‘TCP FIN or TCP RST packets’ corresponds to the teachings surrounding filtering rules in Shieh.” Paper 8, 19. I agree that “Shieh does not appear to teach what happens to a packet in a special event such as a TCP FIN or TCP RST packet is identified” because “Shieh discloses that packet destinations are determined by the bypass flag—which is separate from the TCP FIN or TCP RST packet.” *Id.*

H. Claim 30: The prior art does not disclose “receiving...one or more additional packets” as claimed

111. Claim 30 requires the method of claim 1, further comprising “receiving, by the network node from the first entity over the packet network, one or more additional packets; checking, by the network node, if any one of the one or more additional packets satisfies the criterion” and “responsive to an additional packet not

satisfying the criterion, sending, by the network node over the packet network, the additional packet to the second entity; and responsive to the additional packet satisfying the criterion, sending the additional packet, by the network node over the packet network, in response to the instruction.”

112. I understand Dr. Bhattacharjee’s opinion is that claim 30 is obvious because “Lin teaches that its method can be applied to ‘any packet received by the SDN switch.’ EX1005, Col. 6:40-63. As such, Lin would check if one or more additional packets after the first packet satisfies the packet-applicable criterion for the reasons discussed above for Elements [1.4]-[1.6].” Ex. 1004, ¶198.

113. Lin, Col. 12:59-63 states:

“After the redirect flow rules for creating the SDN pipe are inserted in the flow tables 621, *any packet received by the SDN switch 620 in the ingress port 623-1 will be identified as to be forwarded to the redirect port 623-2*, and *any packet received by the SDN switch 620 in the redirect port 623-2 will be identified as to be forwarded to the ingress port 623-1* (see arrow 602). This allows the security service 630 to receive from the redirect port 623-2 all outgoing packets sent by the sender component 622 to the ingress port 623-1.”

Ex. 1005, 6:40-48.

114. Lin states that “any packet received by the SDN switch 620 in the ingress port 623-1 will be identified as to be forwarded to the redirect port 623-2,

and any packet received by the SDN switch 620 in the redirect port 623-2 will be identified as to be forwarded to the ingress port 623-1.” *Id.* In other words, packets received at ingress port 623-1 are forwarded to redirect port 623-2, and packets received at redirect port 623-2 are forwarded to ingress port 623-1. Lin does not disclose in this passage that the destination of the packets change between the “second entity” and the “entity that is included in the instruction” based on whether the additional packet satisfies the criterion as in claim 30.

115. I understand Dr. Bhattacharjee’s opinion is that claim 30 is obvious because “Shieh discloses that its method can be applied to ‘subsequent packets of a particular session.’ EX1006, ¶[0018]; see EX1006, ¶[0037] (discussing applying the method of Shieh to ‘further packets of the same session’). As such, Shieh would check if one or more additional packets after the first packet satisfies the packet-applicable criterion for the reasons discussed above for Elements [1.4]- [1.6].” Ex. 1004, ¶313.

116. Shieh, paragraph 18 states “[a]lternatively, a security device may inform a network access device that subsequent packets of a particular session should be forwarded from the network access device for security inspection.” Ex. 1006, [0018]. Shieh, paragraph 29 includes a similar statement: “Alternatively, a security device may inform a network access device that subsequent packets of a

particular session should be forwarded from the network access device for security inspection.” Ex. 1006, [0029].

117. In other words, Shieh paragraphs 18 and 29 state that Shieh’s security device informs the network access device that all subsequent packets of the session are forwarded for inspection. Instead of checking whether the additional packet satisfies the criterion and then sending the additional packet to the “second entity” or to the “entity that is included in the instruction” depending on the criterion as in claim 30, paragraphs 18 and 29 of Shieh teach that all packets are forwarded to the same destination (the security device) for inspection. Shieh does not disclose that the “subsequent packets of a particular session” are checked for whether the criterion is satisfied, and then sent to either the “second entity” or the “entity that is included in the instruction” depending on whether the criterion is satisfied.

VIII. CONCLUSION

118. I hereby declare that all statements made herein of my own knowledge are true and all statements made herein on information and belief were and are believed by me to be true, and that all statements herein were and are made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under section 1001 of Title 18 of the United States Code

and that any such willful false statements may jeopardize the validity of the application or any patents issued thereon.

Respectfully submitted,

A handwritten signature in black ink, appearing to read "Miguel Gomez", written in a cursive style.

Miguel Gomez

Dated: January 15, 2024