UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE PATENT TRIAL AND APPEAL BOARD

---

AMAZON.COM, INC.,
Petitioner,

v.

VIRTAMOVE, CORP.,
Patent Owner.

---

Case Nos. IPR2025-000563, IPR2025-00566
Patent No. 7,519,814

---

**DECLARATION OF ERIK DE LA IGLESIA IN SUPPORT OF PATENT OWNER'S PRELIMINARY RESPONSE TO THE PETITION**

I declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true.

Executed on June 10, 2025, at Mountain View, California.

Erik de la Iglesia

## TABLE OF CONTENTS

I, Erik de la Iglesia, hereby declare as follows:

## I. Introduction

1. I am over the age of eighteen (18) years and otherwise competent to make this declaration.

2. I have been retained as an expert witness on behalf of Patent Owner for the above-captioned *inter partes* reviews ("IPRs").[1] I understand that the petitions for *inter partes* review involve U.S. Patent No. 8,519,814 ("the '814 patent").

3. I make this declaration based on my personal knowledge, educational background and training, consideration of the materials I discuss herein, and my expert opinions.

4. My work on this matter is being billed at my normal hourly consulting rate of $750, with reimbursement for actual expenses. My compensation is not related to the outcome of any proceeding involving the '814 patent. I have no personal interest in the outcome of the case.

5. In preparing this Declaration, I have reviewed and considered the '814 patent, the '814 patent's prosecution history, the Petition and prior art references

[1] All citations herein are to the Petition and exhibits corresponding to IPR2025-000563.

submitted with the Petition, the declaration submitted by Dr. Long in this proceeding, and each document cited in my declaration.

## II. Background and Qualifications

6. My qualifications for forming the opinions given in this declaration are summarized here and are addressed more fully in my curriculum vitae, which is submitted as Exhibit 2002.

7. I hold a Bachelor of Science degree in Electrical Engineering from the University of Florida, and a Master of Science degree in Electrical Engineering from Stanford University. While at Stanford, I was an NSF Graduate Research Fellow. My graduate focus included VLSI CMOS architecture, high-speed circuit design, computer networks and protocols, and artificial intelligence. I have worked in several computer and electrical engineering related fields for over 25 years resulting in 68 issued U.S. patents that have been cited over 2850 times in USPTO patent applications. Since 2003, I have served in a number of chief architect and founder roles of companies that have been acquired or gone public. My experiences include the design or implementation of virtualization, concurrent operating system deployment in datacenters and datacenter application deployment and management relevant to the technology of the ’814 patent.

8. Between December 1997 and November 1999, I was a circuit design engineer at Intel Corporation within the Mobile and Handheld Products Group. My

responsibilities included working on several aspects of mobile processors and chipsets including clock and power distribution, dynamic clock frequency selection, and the circuit design, device simulation, and validation of such features. Until March of 2000, I worked on similar functionality within the Microprocessor Products Group for the Itanium 3 and 4 products.

9. My work in power distribution included thermal modeling and predictive control of silicon devices and their packaging such as multi-tiered cooling systems for embedded, mobile, and desktop environments. Such systems included multiple tiers of thermal sensors, active and passive thermal control systems, and the feedback and analytic firmware and software control. My models were part of the development of the Thermal Design Power (TDP) metrics used by OEM and system designers to plan thermal control solutions for consumer and industrial products.

10. Between March 2000 and September 2003, I worked as a logic designer and architect for WebStacks and, through acquisition, Extreme Networks. WebStacks built a fully hardware-based TCP and HTTP processing stack for network processing functions including load balancing, proxy processing, and content rewriting. I was responsible for the architecture and design of the HTTP processing engine including header extraction and processing.

11. Between August 2003 and August 2007, I worked as Founder, Chief Architect and Director of Engineering for Reconnex, later acquired by McAfee. Reconnex built a gigabit line-rate network security analyzer capable of classification, heuristic analysis, and policy enforcement on arbitrary protocols and content types. The Reconnex product captured and analyzed content transmitted over the internet using HTTP and other protocols, including storage protocols. Design of the Reconnex system was based on a modified and customized version of the Linux kernel which included specifically designed driver source code and interfacing between the application and kernel domains. Reconnex technology was employed in the audit and compliance verticals based on its ability to quickly analyze and classify the content of transmissions and interact with protocols and communications crossing corporate and departmental boundaries.

12. Between August 2007 and September 2008, I founded Strangways, a company focused on addressing messaging security for webmail and internet messaging protocols. Products and technology developed by Strangways were sold to SendMail Inc., Iron Port (Cisco), and other companies. The Strangways product utilized a Squid caching server and ICAP content rewrite server to translate HTTP to SMTP.

13. Between September 2008 and April 2014, I worked as Chief Architect for Gridiron Systems and, through acquisition, Senior Director of Technology for

Violin Memory. I held the Senior Director position with Violin Memory both before and after the company's IPO. My responsibilities included architecture and design of flash media controllers, adaptive caching design, and machine learning for application storage behavior analytics. Gridiron Systems products were deployed to enable scaling within virtual environments and to support parallel application deployments requiring concurrent access to large datasets.

14. The Gridiron product was a storage accelerator used in many analytic environments such as big data architectures with constant ingest and processing of financial, consumer and IoT data. In my role as Chief Architect, I often worked with customers to implement and accelerate their analytics environments and spoke on related topics at trade shows and industry user groups. During product development, I worked with the data science team at customer sites to improve database and storage performance for real-time shopping recommendation and user tracking for advertising.

15. Additional information regarding my employment history and prior expert testimony is included in Exhibit 2002. As reflected in Exhibit 2002, I have consulted on several matters involving intellectual property disputes including instances of alleged patent infringement requiring analysis of source code, reviewing and analyzing patents, writing expert reports, and testifying in court. I have testified at deposition and at trial in a number of these cases.

## III. Legal Principles

### A. Claim Construction

16. I understand that the first step in performing a validity analysis of the patent claims is to interpret the meaning and scope of the claims by construing the terms and phrases found in those claims. I understand that the appropriate construction of a claim term is its ordinary and accustomed meaning as understood by one of ordinary skill in the art at the time of the invention in the context of the entire patent and the prosecution history.

17. I understand that standard for claim construction in an *inter partes* review is the same standard as is applied in district court proceedings.

18. I understand that a determination of the meaning and scope of the claims is a matter of law. I have been informed that to determine the meaning of the claims, one should consider the intrinsic evidence, which includes the patent's claims, written description, and prosecution history.

### B. Burden of Proof

19. I understand that in an *inter partes* review, the petitioner has the burden of proving unpatentability by a preponderance of the evidence.

### C. Anticipation

20. I have been instructed by counsel and understand that a reference is anticipated if a single prior art reference discloses each and every claim element,

either explicitly or inherently, as arranged in the same way as in the claim. I understand that where even one claim element is not disclosed in a reference, a contention of anticipation fails.

21. I further understand that when a reference fails to explicitly disclose a claim element, that reference inherently discloses that element only if the reference must ***necessarily*** (or "inherently") include the undisclosed claim element.

**D. Obviousness**

22. I have been instructed by counsel and understand that a combination of prior art references may render a claim obvious if, at the time of the invention, a person of ordinary skill in the art would have selected and combined those prior-art elements in the normal course of research and development to yield the claimed invention.

23. I understand that in an obviousness analysis, one should consider the *Graham* factors, including: the scope and content of the prior art; the differences between the claimed inventions and the prior art; and the level of ordinary skill in the art. I further understand the obviousness analysis is to be performed on a claim-by-claim basis. I understand that a person of ordinary skill in the art is a person of ordinary creativity, not an automaton.

24. I have been instructed by counsel and understand that obviousness requires more than a mere showing that the prior art includes separate references

covering each separate limitation in a claim under examination. I understand obviousness requires the additional showing that a person of ordinary skill at the time of the invention would have been motivated to combine those references in a manner that would include all limitations of the challenged claim, and, in making that combination, a person of ordinary skill in the art would have had a reasonable expectation of success.

25. I also understand that an obviousness analysis must be conducted with awareness of the distortion caused by hindsight bias and with caution of arguments reliant upon *ex post* reasoning. For instance, I understand that when considering obviousness, I should put myself in the position of a person of ordinary skill in the field at the time of the invention, rather than considering new information that is known today, but was not known before the priority date of the challenged patent.

## IV. Person of Ordinary Skill in the Art

26. Dr. Long states that a POSITA in the field of the ’814 patent “would have had a minimum of a bachelor’s degree in computer engineering, computer science, software engineering, or a similar field, and approximately two years of industry or academic experience in a related field” and that “[w]ork experience could substitute for formal education and additional formal education could substitute for work experience.” Ex. 1002 ¶31. For purposes of my analysis in this declaration only, I do not dispute Dr. Long’s proposed level of ordinary skill.

27. I have at least this level of skill in the art, and have had such level of skill since before the September 15, 2003, the earliest alleged priority date of the '814 patent.

## V. Claim Construction

28. I understand that the parties have proposed various claim terms for construction. However, for purposes of the analysis set forth in this declaration, none of those disputes need be resolved.

## VI. I disagree with the Petition's theory as to Ground 1 (based on Osman).

29. Claim 1 of the '814 patent requires, among other things, "storing in memory accessible to at least some of the servers a plurality of secure containers of application software,… wherein the application software cannot be shared between the plurality of secure containers of application software."

30. The Petition alleges that the "cannot be shared" requirement is satisfied, based on the allegation that "private directories" may be created for pods in Osman's disclosure. Pet. 29 (quoting Ex. 1003, 367). According to the Petition, "[t]hese private directories prevent applications running in different pods from interfering with each other." *Id.*

31. In my opinion, the Petition's allegations are in contradiction to the teachings of Osman. Specifically, while Osman does teach private directories, Osman makes clear that private directories in the context of its disclosure do ***not***

"prevent applications running in different pods from interfering with each other." *See* Pet. 29 (citing Ex. 1003, 367). Osman, in fact, teaches that even pods that make ***use*** of private directories may be used to share applications among themselves.

32. Specifically, Osman teaches, in a paragraph devoted to how "[p]rivate pod directories" are used:

> Private pod directories can also be useful for allowing per-pod application configurations without having to duplicate the application file hierarchy. When some files or subdirectories used by a common application need to be specific to a given pod, these files can be easily configured as symbolic links to files in the respective private pod directories. For example, ***to install a web server that is available to all pods***, an administrator could install the web server in a global /usr /local/ apache directory, and make the conf directory within it a symbolic link to /privatepod/apache/conf. ***This will allow multiple pods to share one copy of the web server***, which can be centrally managed and upgraded periodically to fix bugs and close up security holes, while each pod maintains its own configuration, allowing pods to point to log files and web pages anywhere on their file system.

Ex. 1003, 367.

33. In other words, Osman makes clear that multiple pods can share the same application software ***even while using private directories***. Thus, the Petition's reliance on Osman's private directories does not disclose the

obviousness of implementing containers such that "the application software cannot be shared between the plurality of secure containers of application software."

## VII. I disagree with the Petition's theory as to Ground 2 (based on Tucker and Bandhole).

34. Claim 1 of the '814 patent requires that the claimed method be performed "[i]n a system having a plurality of servers with operating systems that differ."

35. In alleging that "operating systems that differ" is satisfied, the Petition relies ***exclusively*** on the Bandhole reference. Pet. 39. In other words, there is no contention that in Tucker, more than a single variation of an operating system could be used.

36. And despite Tucker being the primary reference, neither the Petition nor Dr. Long identify ***any*** supposed motivation to vary Tucker's implementation, such that it would be used with multiple operating systems. The only purported motivation is that the user of ***Tucker's*** technology would improve ***Bandhole***, but they never explain why a POSITA considering Tucker would look to Bandhole in the first place.

37. Furthermore, even if Bandhole and Tucker were considered together, the Petition never explains why the combined system would use multiple different operating systems. Given that Tucker's "zones" indisputably run only on a single

version of Solaris, and given the alleged "security and deployment benefits that Tucker discloses" (Pet. 37), why would a POSITA implementing some combination of Tucker and Bandhole vary the operating systems used in any given deployment? The Petition and Dr. Long provide no answer to this question.

38. Notably, even in the specific example the Petition discusses, there does not appear to be any benefit to deviating from Tucker's disclosure of using only a single operating system. The Petition points to an example in Bandhole where a Solaris server runs certain software, and "[a] separate server using the Linux operating system runs 'Apache web server software.'" Pet. 36-37. But Tucker makes clear (through its incorporation by reference of the Tucker Provisional) that Solaris Zones running the specific compatible version of the Solaris operating system is capable of running Apache web server software. *See* Ex. 1005, page 3 (PDF page 15):

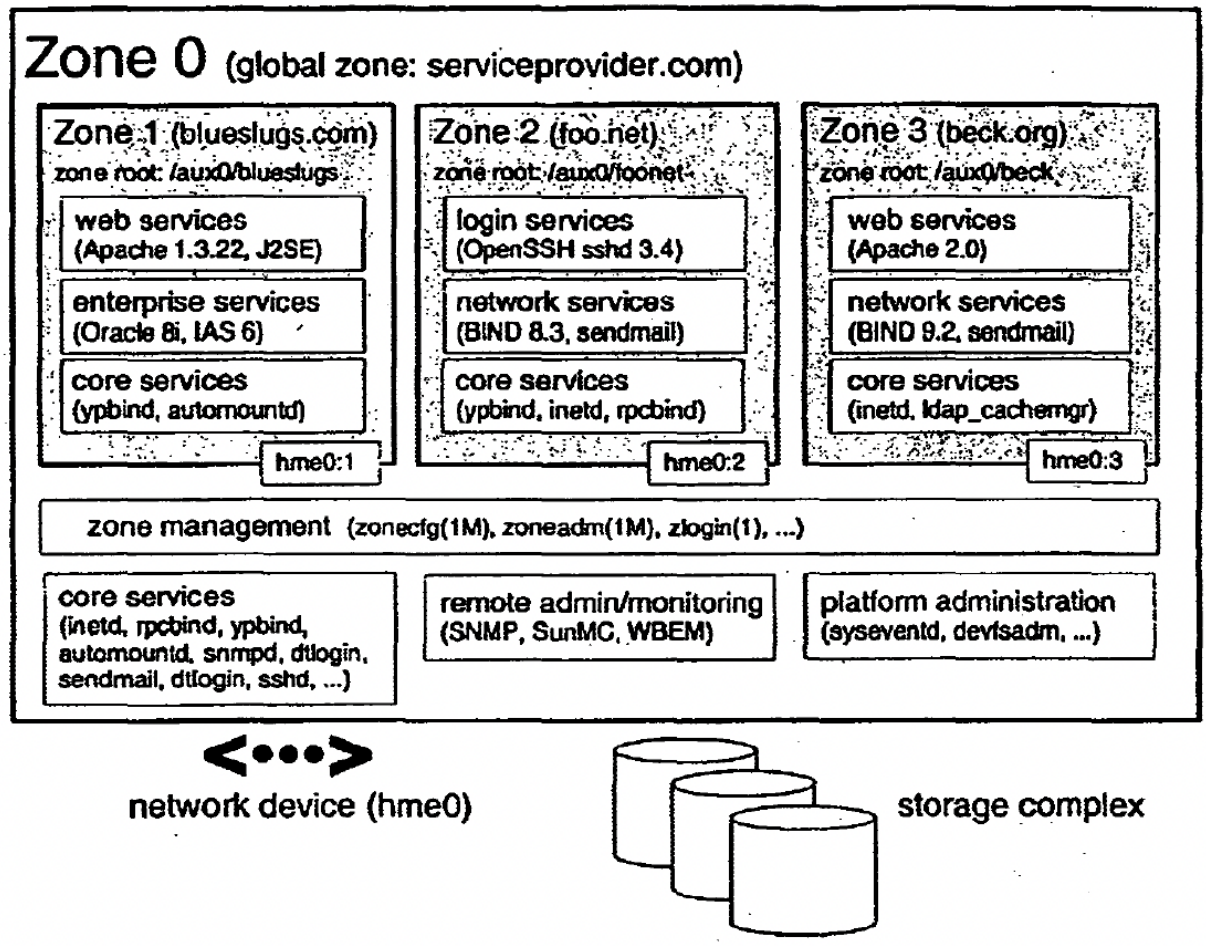


**Figure 1.1: Server Consolidation Using Zones**

39. As shown above, both Zone 1 and Zone 2 are capable of running two different versions of Apache web server software. Given the fact that Solaris was already capable of running the software that the Petition contends would have been run on the Linux operating system in the proposed combination, the Petition presents no plausible reason to use Linux (rather than Solaris) in combination with the Tucker reference to run such software, particularly in view of the benefits the

Petition assigns to Tucker's zones disclosure (which are not even alleged to be compatible with Linux).

40. In sum, in my opinion there would be no motivation to combine Tucker and Bandhole in the way the Petition alleges, because Tucker's zones were compatible only with a specific Solaris OS, and the Petition and Dr. Long identify no reason to use any other operating system in an implementation involving Tucker's zones.

## VIII. I disagree with the Petition's theory as to Ground 3 (based on Gélinas).

41. As noted above, claim 1 of the '814 patent requires that the claimed method be performed "[i]n a system having a plurality of servers with operating systems that differ." Claim 1 ***also*** requires that in the recited method, "the application software cannot be shared between the plurality of secure containers of application software."

42. The Petition relies on different Linux "distributions" as "operating systems that differ" in its alleging that this limitation is satisfied. Pet. 52-53.

43. I note the fact that certain software or certain servers *could potentially be* ***compatible*** with different operating systems does not mean that it would be obvious to actually use that software in a system having different operating systems. I note that, in general, varying the operating systems used within a particular deployed system adds complexity, and a POSITA would need some

reason to implement a varied-operating-system environment to motivate such an approach.

44. The Petition does not specifically explain what the "system having a plurality of servers with operating systems that differ" would look like in its proposed implementation of Gélinas' teachings. *See generally* Pet. 51-53.

45. However, I note that the only evidence that a vserver would actually be used with various distributions within a single system is Gélinas' teaching that "you may want to create several vservers to test[] various distributions." Pet. 52 (quoting Ex. 1007, 31). But this use case is in direct conflict with the Petition's theory as to how limitation 1[a][viii], which recites "the application software cannot be shared between the plurality of secure containers," would be satisfied.

46. Specifically, the Petition acknowledges that Gélinas teaches that "an administrator can configure containers to share files." Pet. 59. And the Petition acknowledges that this is disclosed by Gélinas at page 37, which teaches that in an instance where you "want to create several vservers to test[] various distributions," you would ***also*** "want to share the /home directory between each." Ex. 1007, 31. A POSITA would understand that in this context, any application software in the /home directory would be shared between vservers in the context of Gélinas's disclosure. There is a clear conflict here, such that using vservers to test various

distributions (to satisfy the preamble) involves violating limitation 1[a][viii] under the Petition's theory.

47. Accordingly, the Petition's proposed use case of implementing vservers with different distributions is directly contrary to the Petition's allegation that the implementation would entail preventing the containers of application software from sharing that application software. In short, in my opinion there is no identifiable motivation that a POSITA would have recognized to practice a method that both (1) is in a system having a plurality of servers with operating systems that differ, and (2) satisfies the requirement that "the application software cannot be shared between the plurality of secure containers of application software" as claim 1 of the '814 patent requires.