

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

AMAZON.COM, INC.,
Petitioner,

v.

VIRTAMOVE, CORP.,
Patent Owner.

Case No. IPR2025-000566
Patent No. 7,519,814

PATENT OWNER'S PRELIMINARY RESPONSE

TABLE OF CONTENTS

I. INTRODUCTION 1

II. ARGUMENT 3

 A. The Petition’s Ground 1 (Osman) 3

 B. The Petition’s Ground 2 (Tucker and Bandhole) 4

 C. The Petition’s Ground 3 (Gélinas) 8

III. CONCLUSION 10

PATENT OWNER'S EXHIBIT LIST

No.	Description
2001	Declaration of Erik de la Iglesia in Support of Patent Owner's Preliminary Response to Petition
2002	<i>Curriculum vitae</i> of Erik de la Iglesia

I. INTRODUCTION

Patent Owner will disclaim claims 31-34 of the '814 patent shortly after the filing of this Preliminary Response, leaving claims 3, 5, 7, 7, 11-12, and 15-16 (all of which depend from claim 1) as the only remaining claims challenged by this Petition.¹ Because the Board may not institute an *inter partes* review based on disclaimed claims,² institution of this proceeding is appropriate only if the Petition can demonstrate a reasonable likelihood of prevailing with respect to claim 1 (from which all other challenged claims depend), the sole remaining challenged independent claim.

The Petition presents three distinct grounds challenging claim 1, each of which fails. The Petition's Ground 1 (involving Osman) fails because it relies on a theory wherein Osman's "private directories" disclose the claim 1 requirement that "the application software cannot be shared between the plurality of secure containers of application software." But Osman makes clear that private directories do *not*

¹ Patent Owner does not concede the merits of the Petition's challenges as to any claims; disclaimer of the referenced claims is merely to simplify proceedings.

² 37 C.F.R. § 42.107(e) ("No *inter partes* review will be instituted based on disclaimed claims.").

prevent sharing of application software, and indeed even discloses how those private directories are utilized to share application software between the alleged containers.

The Petition's Ground 2 (involving Tucker and Bandhole) also fails. Tucker is not alleged to disclose "a plurality of servers with operating systems that differ" as recited by claim 1, and a POSITA would understand that Tucker's teachings of "zones" are compatible with only a single operating system. And the Petition provides no motivation whatsoever to modify Tucker's teachings such that Tucker would be implemented *within* "a system having a plurality of servers with operating servers that differ" as claim 1 requires.

The Petition's Ground 3 (involving Gélinas) fails because, like Tucker, Gélinas does not disclose a "a system having a plurality of servers with operating systems that differ" as claim 1 requires. Furthermore, the teaching in Gélinas that the Petition relies on as allegedly disclosing that requirement is inconsistent with the requirement that "the application software cannot be shared between the plurality of secure containers of application software," because Gélinas teaches that in a scenario where the alleged containers might be used to test varying "distributions," any application software in the /home directory would be *shared* between each of the alleged containers. *See* Ex. 1007, 31 ("For example, you may want to create several vservers to test[] various distributions, yet *you want to share the /home directory between each*.... This is probably the easiest way to share data between vservers.").

Accordingly, Patent Owner respectfully requests that institution be denied.

II. ARGUMENT

A. The Petition's Ground 1 (Osman)

Claim 1 of the '814 patent recites, *inter alia*, “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.”

Under the theory of the Petition, 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.*

However, Osman is clear that in the context of its teachings, private directories do *not* as the Petition contends “prevent applications running in different pods from interfering with each other.” See Pet. 29 (citing Ex. 1003, 367). Ex. 2001, ¶¶31-33.

To the contrary, Osman teaches that even pods that make use of private directories may be used to share applications among themselves. 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.

In other words, Osman is *explicit* that multiple pods can share the same application software *even while using private directories*, such that it is no matter that Osman uses private directories by default. Ex. 2001, ¶33. 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.” Ex. 2001, ¶¶29-33.

B. The Petition’s Ground 2 (Tucker and Bandhole)

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.”

In alleging that “operating systems that differ” is satisfied, the Petition relies *exclusively* on the Bandhole reference. Pet. 45. In other words, there is no contention that in Tucker, more than a single variation of an operating system would (or even could) be used. *See generally id.*; *see also id.* at 5 (“Sun Microsystems was working on containers *for its popular Solaris operating system.*”); *id.* at 6 (“Solaris Zones are also secure containers that isolate and confine processes *that share an underlying operating system.*”).

And despite Tucker being the primary reference, the Petition fails to 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 use of *Tucker’s* technology would improve Bandhole (*see* Pet. 43-44), but the Petition never explains why a POSITA considering *Tucker* would look to *Bandhole* in the first place. In other words, there is no allegation that the combination of Tucker with Bandhole accomplishes anything more than Tucker, standing alone. *See* Ex. 1001, ¶¶35-36.

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,” the Petition fails to provide any reason why a POSITA would

vary the operating systems used in a system implementing Tucker’s teachings. Ex. 2001, ¶¶37-40.

Notably, even in the *only* example of using multiple operating systems in Bandhole that the Petition actually evaluates, there is *no* alleged to be any benefit to deviating from Tucker’s disclosure of using only a *single* operating system. Specifically, 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. 42-43. 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.

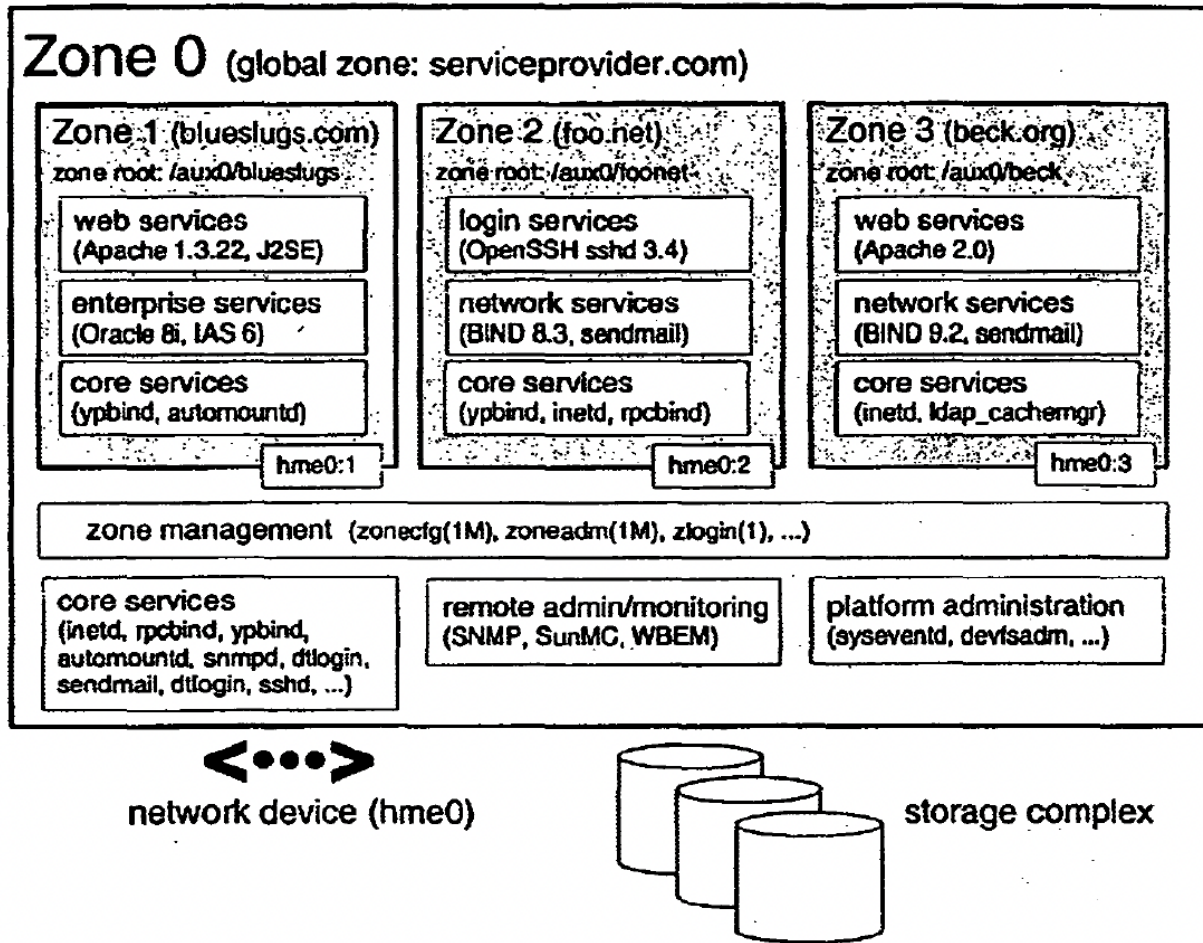


Figure 1.1: Server Consolidation Using Zones

Ex. 1005 (Tucker Provisional) at page 3; Ex. 1004 at 1:6-10 (incorporating provisional by reference); Ex. 2001, ¶38.

As shown in the incorporated-by-reference provisional figure 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). Accordingly, the Petition fails to identify any reason to use multiple operating systems in the context of a combination with Tucker. Ex. 2001, ¶¶39-40.

C. The Petition's Ground 3 (Gélinas)

As discussed 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.”

The Petition relies on different Linux “distributions” as “operating systems that differ” in its alleging that this limitation is satisfied. Pet. 56. As explained by Patent Owner's expert, whether software *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, because varying the operating systems used within a system has the costs of added complexity. Ex. 2001, ¶¶41-43. 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. But the

only specific example the Petition relies on from Gélinas that *allegedly* teaches a single system utilizing multiple operating systems is Gélinas' teaching that "you may want to create several vservers to test[] various distributions." Pet. 56 (quoting Ex. 1007, 31).

As an initial matter, testing various distributions does not necessarily mean that each of these distributions are part of a single system. But even assuming that to be the case, the use case in Gélinas that the Petition relies upon 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. *See* Ex. 2001, ¶¶45-48.

Specifically, the Petition acknowledges that Gélinas teaches that "an administrator can configure containers to share files." Pet. 62. And the Petition acknowledges that this is disclosed by Gélinas at page 37, which teaches that in an instance where a user "want[s] to create several vservers to test[] various distributions," that user would *also* "want to share the /home directory between each." Ex. 1007, 31 (cited by Pet. 62). 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. Ex. 2001, ¶46. The Petition thus fails to articulate a specific use case that satisfies *both* the "operating systems" that differ limitation, *as well as* the requirement that "application software cannot be shared

between the plurality of secure containers” under the Petition’s theory. Ex. 2001, ¶¶46-47.

III. CONCLUSION

For the foregoing reasons, Patent Owner respectfully requests that institution be denied.

Date: June 20, 2025

Respectfully submitted,

/James A. Milkey/
James A. Milkey, Reg. No. 79,503
RUSS, AUGUST & KABAT
12424 Wilshire Blvd., 12th Fl.
Los Angeles, CA 90025

Counsel for Patent Owner

CERTIFICATION REGARDING WORD COUNT

Pursuant to 37 C.F.R. §42.24(d), Patent Owner certifies that there are 1,811 words in the paper excluding the portions exempted under 37 C.F.R. §42.24(a)(1).

Date: June 20, 2025

Respectfully submitted,

/James A. Milkey/
James A. Milkey, Reg. No. 79,503
RUSS, AUGUST & KABAT
12424 Wilshire Blvd., 12th Fl.
Los Angeles, CA 90025

Counsel for Patent Owner

CERTIFICATE OF SERVICE (37 C.F.R. § 42.6(e))

The undersigned hereby certifies that the above document was served on April June 20, 2025 by filing this document through the Patent Trial and Appeal Case Tracking System (PTACTS) as well as delivering a copy via electronic mail upon the following attorneys of record for Petitioner:

Colin B. Heideman (Reg. No. 61,513)
Joseph R. Re (Reg. No. 31,291)
Jeremy A. Anapol (Reg. No. 75,686)
Nathan D. Reeves (Reg. No. 77,806)
Logan P. Young (Reg. No. 79,294)
Knobbe, Martens, Olson & Bear LLP
BoxSEAZN2L2102LP1@knobbe.com

Date: June 20, 2025

Respectfully submitted,

/James A. Milkey/
James A. Milkey, Reg. No. 79,503
RUSS, AUGUST & KABAT
12424 Wilshire Blvd., 12th Fl.
Los Angeles, CA 90025

Counsel for Patent Owner