

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

PALO ALTO NETWORKS, INC.

Petitioner

v.

BT AMERICAS INC.

Patent Owner.

Case IPR2023-00889
Patent 7,895,641

PATENT OWNER'S PRELIMINARY RESPONSE

TABLE OF CONTENTS

I.	INTRODUCTION	1
II.	BACKGROUND ON THE INVENTORS AND ASSIGNEE.....	5
III.	THE ‘641 PATENT	6
	A. The ‘641 Patent’s Disclosure	6
	B. Petitioner Failed to Address Secondary Considerations Previously Raised	12
IV.	DISCUSSION OF ASSERTED REFERENCES	13
	A. Duvall	13
	1. Transmitting Content with TCP/IP	15
	2. Duvall’s Filter Actions.....	16
	3. Other Aspects of Duvall	23
	B. Chu	24
	1. Chu Focuses on Trust Management	25
	2. Chu’s REFEREE Trust Management System	26
	3. Chu’s Sample REFEREE Policy	27
	C. Cogger	28
V.	LEVEL OF SKILL IN THE ART	31
VI.	CLAIM CONSTRUCTION.....	32
	A. “Post-Filtering Residue”	32
VII.	THE PETITION FAILS TO ESTABLISH A REASONABLE LIKELIHOOD THAT THE ASSERTED REFERENCES RENDER ANY CLAIM UNPATENTABLE	35
	A. Petitioner Fails to Meet Its Burden	35
	B. Multiple Features Present in Every Claim Are Missing from Petitioner’s Asserted References	36
	1. Duvall and Chu Lack Analysis of Post-Filter Residue Status Data to Identify Potential Security-Related Events	37

2. Duvall and Chu Lack a Second-level of Analysis By an Analyst System.....	42
3. The Limitations Regarding Receipt of Feedback at the Probe to Dynamically Enhance Its Analytical Capabilities Are Missing from the Asserted References	47
C. There Is No Basis for Combining Duvall and Chu, No Details How to Do So, and No Reasonable Expectation of Success	51
1. Duvall Is Unsuitable for Use as a Network Security Monitoring System.....	53
2. Petitioner’s Motivation Relies on a Mischaracterization of Duvall’s Concerns about Subjectively “Filtering Too Much or Too Little”	58
3. Duvall and Chu Use Filters in Different Ways and for Different Purposes	59
4. A POSA Would Not Combine Chu with Duvall Given the Differences In Concerns, Underlying Assumptions, and Goals...60	
5. Even If a POSA Was Motivated to Search for Techniques to Resolve the Status of Residue, Chu Does Not Teach Any Such Techniques.	61
6. A POSA Would Not Have Arrived at the Claims Given Duvall and Chu.	62
D. There Is No Way to Combine Cogger into Duvall-Chu To Arrive At Independent Claim 18	67
VIII. CONCLUSION	69
IX. WORD COUNT CERTIFICATION	71
X. CERTIFICATE OF SERVICE	72

IPR2023-00889

U.S. Patent No. 7,895,641

PATENT OWNER'S LIST OF EXHIBITS

EX2001 – Declaration of Wenke Lee

EX2002 – Declaration of Bruce Schneier from IPR2019-01324

EX2003 – Bruce Schneier's Wikipedia

EX2004 – John Callas' Wikipedia

I. INTRODUCTION

The present petition represents the second attempt of a party to institute *inter partes* reexamination of U.S. Patent No. 7,895,641 (“’641 Patent” or “Schneier”). Notably, the PTAB already rejected a challenge of the ’641 Patent based on references lacking the same limitations as are missing here, using art that was far closer than what Petitioner now asserts. *See* IPR2019-01324, Paper 9 (denying institution). The present Petition should be denied for the same reasons as the prior denied petition.

Schneier describes a novel architecture for network security monitoring, broken out at the highest level into a probe (EX1001, Fig. 1, 2000) and a Secure Operations Center (“SOC”). This architecture enables the identification of and response to security-related events created by a threat that may have been entirely unknown. *See, e.g.*, EX2001, ¶¶21, 27, 41, 72. After the probe selects and discards status data related to known events through positive and negative filtering, Schneier then performs a two-level analysis on the post-filtering residue. First, the probe analyzes the leftover indeterminate status data (the residue) to identify events rising to the level of a *potential* security event and then sends information about what was identified to analyst systems located at the SOC for the second-level analysis (to confirm whether they are *actual* security events). Each of Schneier’s independent claims include specific claim limitations directed to these

features. Schneier claim 1 also includes limitations related to features for dynamically updating the probe based on the empirical second-level analysis provided by the SOC, to capture the previously unknown threats and render them as known threats for the next time.

Petitioner contends Schneier's independent claim 1 would have been obvious to a person of ordinary skill in the art ("POSA") in view of prior art references Duvall and Chu. But neither reference, nor the alleged combination, discloses the novel features claimed by Schneier. This is not surprising because neither reference relates to the field of network security monitoring—with Duvall directed to censoring subjectively offensive Internet content and Chu directed to confirming the origin of files as coming from a trusted source or not. Indeed, even between each other, the relied-upon references are related to different fields. But, putting those differences aside, both Duvall and Chu only filter based on criteria already known to the user. Unlike Schneier, neither reference has the capability to identify previously unknown threats. The combination fares no better.

In attempting to apply Duvall to Schneier, Petitioner artificially divides Duvall's lone filtering process into two parts, notwithstanding that Duvall only discloses the application of a single set of filters. Petitioner attempts to map a first portion of Duvall's filters to the claimed positive and negative filters of Schneier and a second portion of the same Duvall filters to Schneier's two-step, post-

filtering residue (unknown content) analysis. But both portions of the Duvall filters are part of the same process Duvall uses to positively and negatively filter known (rather than potential/unknown) content. Indeed, these filters are all just known entries in Duvall's filtering database, which Duvall applies in sequence, in the predetermined order, until it finds a first match. Duvall never analyzes post-filtered unknown data (residue), let alone analyze that data twice as Schneier does.

Petitioner's attempt to match Chu's "user intervention" to Schneier's second-level confirmation step fails no better. Chu only teaches "user intervention" by the original application, but not an analyst system. Chu discloses no analysis of data that could not be matched to known criteria (i.e., unknown) after filtering and certainly never transmits such "unknown" or potential data for analysis. And of course, it's nonsensical to add a confirmation step to Duvall when it only identifies material already known rather than *potential* events needing confirmation. Why would you need to confirm that which was already known? The Petition is silent on this point.

Relative to Schneier's analyst system and dynamic updating limitations, Petitioner fails to identify any teaching from either reference whatsoever. Instead, Petitioner improperly relies on conclusory statements from its expert—driven by improper hindsight. Modifying Duvall and Chu to include an analyst system and dynamic capabilities would not have been obvious from either the disclosures of

these references themselves or in light of the secondary considerations. And neither reference discloses a feedback loop so that threats that were previously unknown could be treated as known next time.

Neither reference teaches or discloses any of these essential limitations from the independent claims. So, there is no way to fairly combine these references to recreate Schneier's inventive system.

Fundamentally, even if the needed functionalities were present (and they are not) a POSA simply would not combine a device focused on censoring content (Duvall) with a trust management device wholly unconcerned with detecting network threats (Chu) to create a security monitoring device to identify unknown network threats (Schneier). Though both asserted prior art references each have filters (as does Schneier), that is the extent of their similarity. They are focused on entirely different things and each use their filters differently. Petitioner's sole motivation to combine is based on the expert speculation that uncertainties about what should be censored by Duvall could be resolved through objective analysis techniques of Chu. But that is nonsensical because Duvall intentionally censors based on a user's subjective preferences rather than objectively identifying threats. In other words, there is no reason to believe Chu is capable of determining what is subjectively objectionable to a user of Duvall. And regardless, Chu does not even teach the techniques that Petitioner argues a POSA would search for.

Not only is Petitioner's Duvall-Chu combination nonsensical, but adding Cogger to the mix makes even less sense. Cogger only discloses a remote interface for accessing pre-existing trouble-ticketing systems used by telecommunications service providers. Cogger is irrelevant to Duvall, Chu, and network security. And Cogger's disclosure is wholly inadequate to show how or why a POSA would modify Duvall-Chu as to read on Schneier's independent claim 18.

Because neither of the references, alone or in combination, teach (1) a first-level analysis at the probe of post-filtering residue to identify *potential* (unknown) security-related events after the completion of positive and negative filtering of known events; (2) transmitting information identified from the first-level analysis to an analyst system (for a second-level confirmatory analysis); or (3) dynamic modification of the analysis capabilities of the probe (based on the empirical second-level analysis), the PTAB should deny institution of the petition.

II. BACKGROUND ON THE INVENTORS AND ASSIGNEE

The '641 Patent arose from a security startup, Counterpane, at the dawn of the development of monitored network security services. EX2002. The inventors are widely recognized as luminaries. *Id.*, ¶¶2, 4; *see* EX2003; EX2004.

The architecture described was a major step forward in the field of intrusion detection at a time when there had been no recent breakthroughs. EX2002 ¶¶5-8. The technology Counterpane developed, disclosed and claimed in the '641 Patent,

spawned an entirely new type of managed security monitoring service sufficiently impressive that BT bought Counterpane in 2006—merging it into BT Americas Inc., which now holds title to Counterpane’s patents. EX2002 ¶¶5-8.

III. THE ‘641 PATENT

A. The ‘641 Patent’s Disclosure

Schneier relates to “network security and, more specifically, to methods and systems for dynamic network intrusion monitoring, detection and response.”

EX1001, 1:18-20. Schneier discloses a managed security monitoring system for monitoring, detecting and responding to network intrusions in a customer’s network. *Id.*, 1:57-2:2. Figure 1 provides an overview:

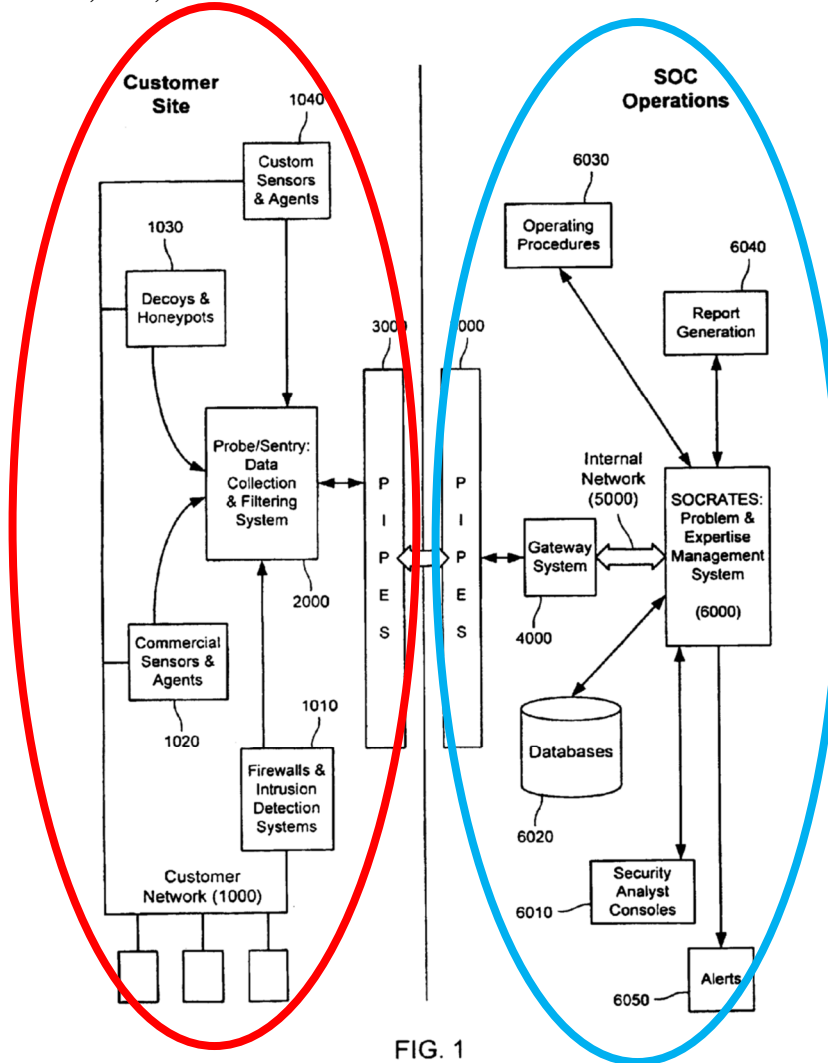


FIG. 1

Fig. 1 depicts a customer site (circled in red) and a Secure Operations Center (the SOC) (circled in blue). *Id.*, 4:45-49. Pipes 3000 provide a secure communications pathway between the customer site and the SOC. *Id.*, 5:50-54. A probe 2000, located at the customer site, collects and filters status data extracted or derived from traffic—with sensors in the form of firewalls, intrusion detection systems 1010, commercial or custom sensors or agents (1020, 1040), and decoys

and honeypots 1030. *Id.*, 4:59-64. The probe monitors and collects information from any component providing status data. *Id.*, 4:64-5:3.

Table 6 of the '641 Patent provides examples of "status data," but the term is not limited to those fields identified. EX1001, 9:2-6. Status data is "data extracted from or generated about the traffic or system processing it that is informative as to the status of the network and its components." EX1013 at 1.

A filtering subsystem within probe 2000 filters status data into three resulting categories: positively filtered data, negatively filtered data, and "post-filtering residue." EX1001, 8:47-62. After filtering, the status data not selected by positive filtering or discarded by negative filtering, is analyzed by an anomaly engine (EX1001, Fig. 2, 2050) to search for potentially security-related events involving previously unknown threats. EX2001, ¶¶53, 72-73. Thus "post-filtering residue, wherein the post-filtering residue is data neither discarded nor selected by filtering" is "status data that undergoes negative and positive filtering, but is neither discarded by such negative filtering nor selected by such positive filtering." EX1013 at 4.

Figure 2, reproduced below, describes the probe in greater detail than Figure 1:

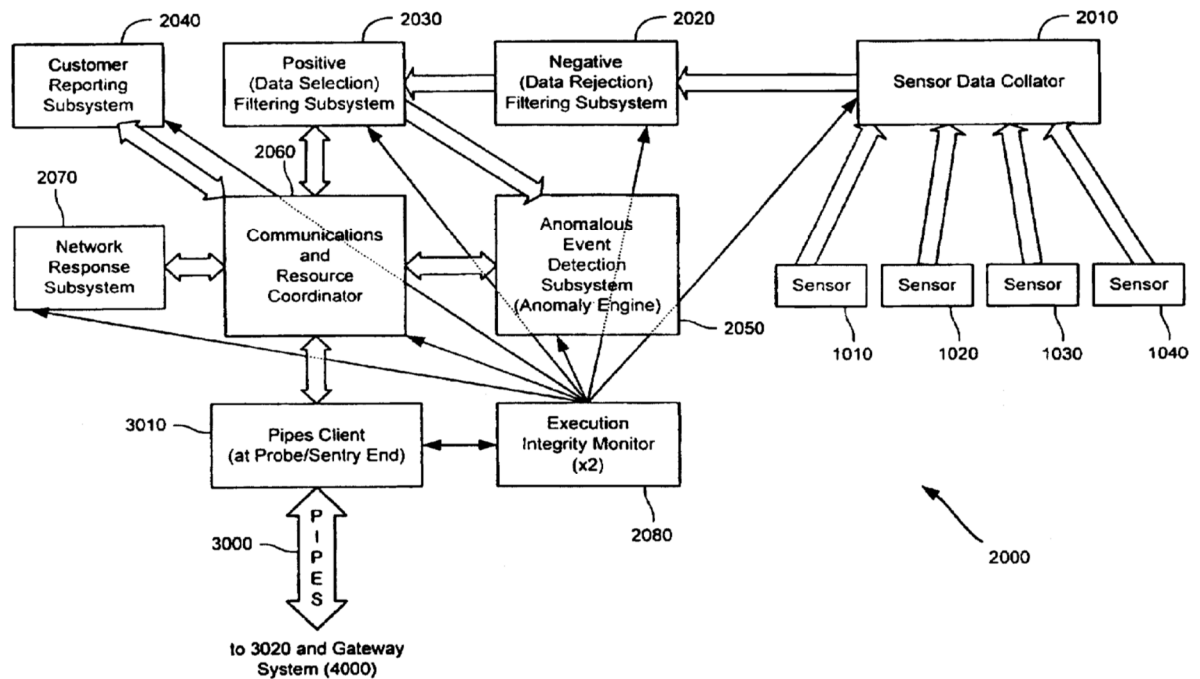


FIG. 2

In operation, sensor data collator 2010 collates status data collected by the sensors. Then:

Once collated, the data is first filtered by negative filtering subsystem 2020, which discards uninteresting information, and then by positive filtering subsystem 2030, which selects possibly interesting information and forwards it to communications and resource coordinator 2060. Data neither discarded by negative filtering subsystem 2020 nor selected out as interesting by positive filtering subsystem 2030 form “residue,” which is sent to anomaly engine 2050 for further analysis.

EX1001, 8:51-59.

Anomaly engine 2050 picks through the residue to determine what residue information is sufficiently interesting to rise to the level of a potentially security-related event. EX1001, 8:59-62; EX2001, ¶¶53, 72-73.

Therefore, the probe decides, after filtering, based on an analysis by anomaly engine 2050 of leftover status data (residue), whether there might be an event worth investigating. EX2001, ¶¶51-55, 143-145. The analysis of leftover residue flags potential security-related events to control the amount of information subsequently transmitted to the SOC. *Id.*, ¶¶73. Once a potentially security-related event is identified, information about it is sent to a SOC (EX1001, 3:17-21) where trained security analysts using analyst systems, having access to additional tools and databases, can make an informed determination as to whether there has been an event or how best to respond to the potential event (EX1001, 3:41-62).

Specifically, information about an event identified by anomaly engine 2050 is sent to resources coordinator 2060, creating sentry messages transmitted to the SOC through pipes 3000. EX1001, 8:55-13. The “[c]ommunications and resource coordinator 2060 creates sentry messages out of the interesting status data and forwards those messages on to gateway system 4000.” *Id.*, 8:66-9:1. Gateway system 4000 is broken out schematically in Figure 3 and described at EX1001, 9:42-53. Sentry update manager 4010 is responsible for updates sent back to the

probe and a communications and management coordinator 4020 converts sentry messages into gateway messages passed onto the SOC. *Id.*

The analysts at the SOC are aided by SOCRATES (Fig. 1, 6000; Secure Operations Center Responsive Analyst Technical Expertise System per EX1001, 3:61-62), which manages and addresses actual and potential problems at the customer site (*e.g.*, as reported through sentry messages generated from an analysis of residue status data). SOCRATES and its associated databases and consoles are shown in Figure 4. EX2001, ¶¶63.

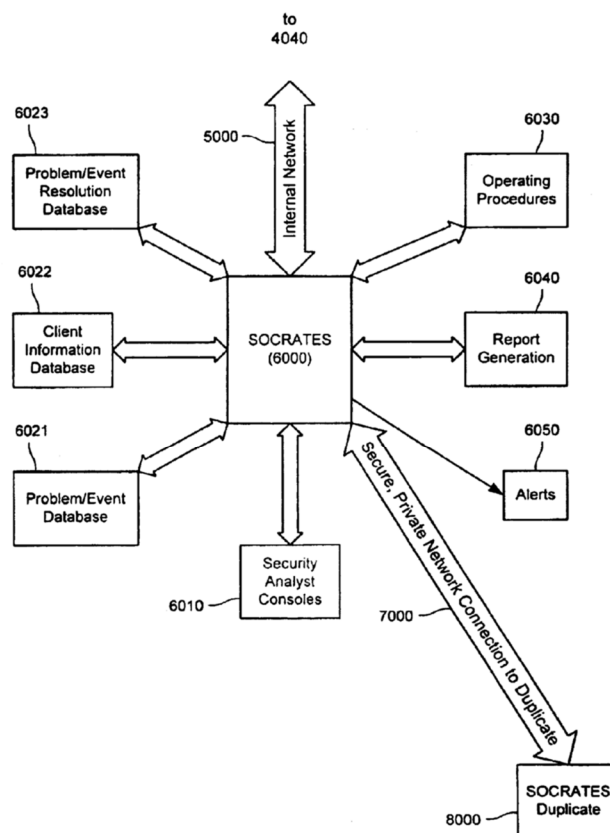


FIG. 4

As reflected in Figure 4, security analyst consoles 6010, which interface with SOCRATES, are available to security analysts, *i.e.*, “personnel specializing in the analysis of network attacks.” EX1001, 2:43-44. In an exemplary embodiment, the security analyst works in concert with SOCRATES to categorize, prioritize, investigate, and respond to customer incidents or problems. *Id.*, 3:42-45.

SOCRATES has access to multiple databases, shown as 6021, 6022, and 6023, which provide “Security Intelligence” in the form of characteristics and known vulnerabilities of certain products, characteristics of hacker tools, and the effectiveness of alternative responses. *Id.*, 2:51-64; EX2001, ¶65.

Once that second analysis is complete, the system provides feedback from the SOC to the probe where resource coordinator 2060 updates software at the probe, dynamically. EX1001, 5:32-41. Feedback received at the probe updates the analytical capabilities of the probe “during actual operation, rather than offline.” EX1013 at 2; EX2001, ¶68.

B. Petitioner Failed to Address Secondary Considerations Previously Raised

Patent Owner previously raised secondary considerations in response to a prior petition against the '641 Patent. They apply equally here. Petitioner failed to address any of these secondary considerations despite the Board's repeated cautioning that “known evidence of secondary considerations should be addressed

in the Petition.” *Robert Bosch Tool Corp. v. SD3, LLC*, IPR2016-01753, Paper 15 at 28 (PTAB Mar. 22, 2017) (collecting cases denying petition for failing to address objective indicia when present).

For example, the Schneier inventors recognized that clues about unknown threats were most likely to reside in the residue left after positive and negative filtering, which was of little interest at the time. EX2001, ¶¶72. The initial analysis at the probe cut down considerably on the flow of traffic to the SOC, whereas prior art approaches transmitted all traffic/information or discarded the residue. *Id.*, ¶¶73. And the second-level analysis of the claimed architecture allowed the system to reduce the possibilities of false positives by giving analysts access to more information and tools (*e.g.*, across probes). *Id.*, ¶¶74-75.

The PTAB should consider the advancements in the art enabled by the claimed inventions as unrebutted indicia of non-obviousness.

IV. DISCUSSION OF ASSERTED REFERENCES

A. Duvall

Duvall focuses on censorship by filtering “personally objectionable” “material received over the Internet.” *See* EX1004, 2:12-13. Personally objectionable material is subjectively pre-determined “depending on [each] user’s individual tastes and sensitivities.” *Id.*, 2:12-16. Duvall explains many “parents

are particularly concerned that their children can get access to [] objectionable materials,” such as sexually explicit material. *Id.*, 1:13-17.

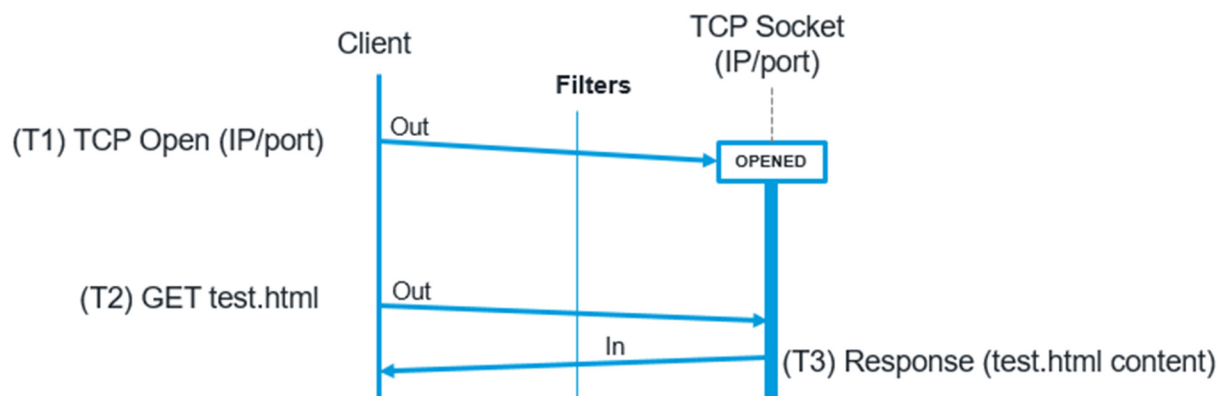
Duvall provides filters and various filter-actions for censoring to accommodate users’ different preferences. *See id.*, 2:12-16. Duvall contemplates two types of users: users with privilege to configure the database (*e.g.*, parents/companies), and end-users lacking those privileges (*e.g.*, children/employees) to whom the censorship is meant to apply. *See, e.g., id.*, 8:1-16, 11:18-20; *see also* EX2001, ¶79.

According to Duvall, the filtering system “compares portions of incoming and/or outgoing messages to filtering information in a filter database and determines whether to block or allow incoming and/or outgoing transmissions of messages in response to the comparison.” EX1001, Abstract. All filters are represented as entries in a common database, with fields for criteria and fields specifying an action to take. *See* 3:64-4:15; 5:8-19. Objectionable material is blocked “[i]n response to a match” with a single filter from the database (*id.*, Abstract; *id.*, 1:35-40), which “takes action based on the first matched filter” (*id.*, 4:29-30). Specifically, Duvall discloses three blocking actions available when filters match specified content: “discarding incoming data, preventing execution of an open command, or replacing parts of received data.” *Id.*, 1:35-40. In other

words, Duvall teaches a single filter system the user can configure to match known criteria, along with several filter actions that can be taken when there is a match.

1. Transmitting Content with TCP/IP

Duvall discloses a filtering system operating on TCP/IP traffic. Duvall explains that “[m]essages are transmitted over the Internet ... with a transport protocol known as Transmission Control Protocol/Internet Protocol (TCP/IP).” *Id.*, 2:41-44. To access websites, URLs are resolved into IP addresses (“IP”) and ports, which a client uses to open a TCP connection with a host. *Id.*, 2:66-3:19. This TCP data stream allows the client to communicate over the Internet with the website host at that IP/port. *Id.*, 2:58-62. Data from the website can then be requested and received through the TCP data stream. *See id.*, 3:18-19; 6:24-25. The following sequence diagram shows how messages are sent in Duvall’s detailed example:



EX2001, ¶86.¹

After resolving a URL's IP/port, the client sends an initial outgoing transmission (T1) to open the TCP connection (*i.e.*, TCP Socket). *See* EX1004, 2:66-3:19. Once opened, the client requests the desired webpage through the opened TCP data stream in an outgoing transmission (GET test.html, T2). *See id.*, 3:18-19; 6:24-25. The host responds and transmits the requested content back to the client in an incoming transmission (test.html content, T3) via the same stream. *See id.*

Duvall monitors these incoming and outgoing transmissions (*i.e.*, T1-T3). *Id.*, 3:50-63. After matching the transmission to a filter, Duvall takes the action specified by that filter—*i.e.*, blocking action or allowing action. *See id.*, 1:34-40. Notably, for each sequence of transmissions and responses, shown above, Duvall attempts to apply the same sequence of filters in sequence until one matches. *Id.*, 4:29-30.

2. Duvall's Filter Actions

Duvall's filtering system censors content based on users' personal tastes and sensitivities. *See id.*, 2:12-16. Users subjectively define what they want to censor

¹ The sequence diagrams used herein are created for illustrative purposes based on Duvall's detailed example.

by specifying the matching criteria of the filters (either the known objectionable content itself, or the known sources of such content). *See id.*, 6:60-7:8. Matched content can then be censored with Duvall’s “blocking actions” or allowed. *Id.*, 1:34-40.

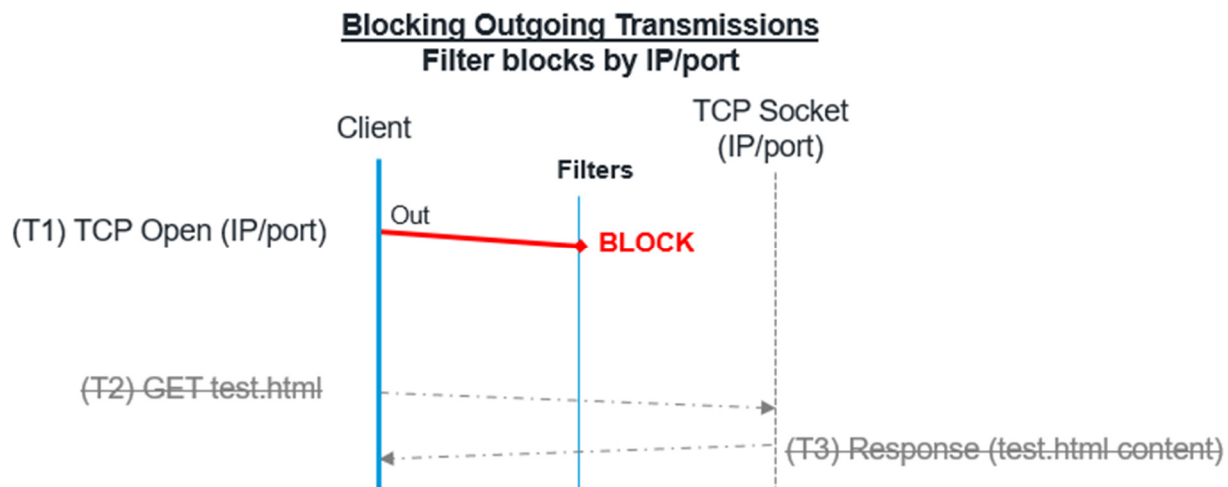
Duvall’s three blocking actions enable the same set of filters to work on incoming or outgoing traffic and block specific objectionable content in the traffic or known sources of objectionable content. *See id.*; *id.*, 4:22-28. “Blocking actions that are available for incoming messages are different from those for outgoing messages.” *Id.*, 5:35-37. Outgoing transmission that would otherwise open the data stream can be blocked when the IP address is known to be a source of objectionable content—unconditionally censoring communication with the host by “preventing execution of an open command.” *Id.*, 1:36-40; *see* 4:56-61; 6:14-18; *see also* EX2001, ¶90. For incoming transmissions, Duvall can “discard[] incoming data ... or replac[e] parts of received data” to censor specific pages or specific words/content on a page the user finds objectionable. EX1004, 1:36-40; *see id.*, 5:35-40; EX2001, ¶90.

Duvall categorizes its filter actions into two logical categories—direct and deferred-action. Notably, both only block/allow known objectionable content or known sources of objectionable content. But they are described as applying to the sequence of TCP/IP transmissions to censor in different ways (summarized below).

Neither the filters nor the filter-actions ever identifies anything previously unknown or elevates it to the level of a “potential” event. There is also no feedback loop to update the system dynamically to capture the previously “unknown” as “known” next time.

a. Blocking All Traffic From a Source

Duvall can censor a domain known to be a source of objectionable content by blocking initial outgoing transmissions used to open TCP connections based on the IP. *See* EX1004, 1:36-40. A visualization is provided below:



EX2001, ¶94.

Duvall prevents the TCP stream from opening by blocking the transmission to open the stream (*i.e.*, T1 never reaches destination). *See* EX1004, 4:56-58.

Without an open stream, content cannot be requested or received from the host (*i.e.*, request (T2) cannot be sent, and host cannot respond (T3)). EX2001, ¶95.

All content from this IP/port is unconditionally censored because no TCP connection is ever made. *See* EX1004, 4:38-55.

Alternatively, a filter's action can be to allow all communications with an IP/port (*i.e.*, by allowing the data stream to be opened). *See id.* This lets the client request and receive all content from that stream unconditionally. EX2001, ¶96. These BLOCK/ALLOW actions are “direct-actions” “carried out immediately” on the matched transmission and they “indicate that the system should unconditionally allow or unconditionally block the transmission.” EX1004, 4:15-20.

b. Blocking Specific Content From a Source

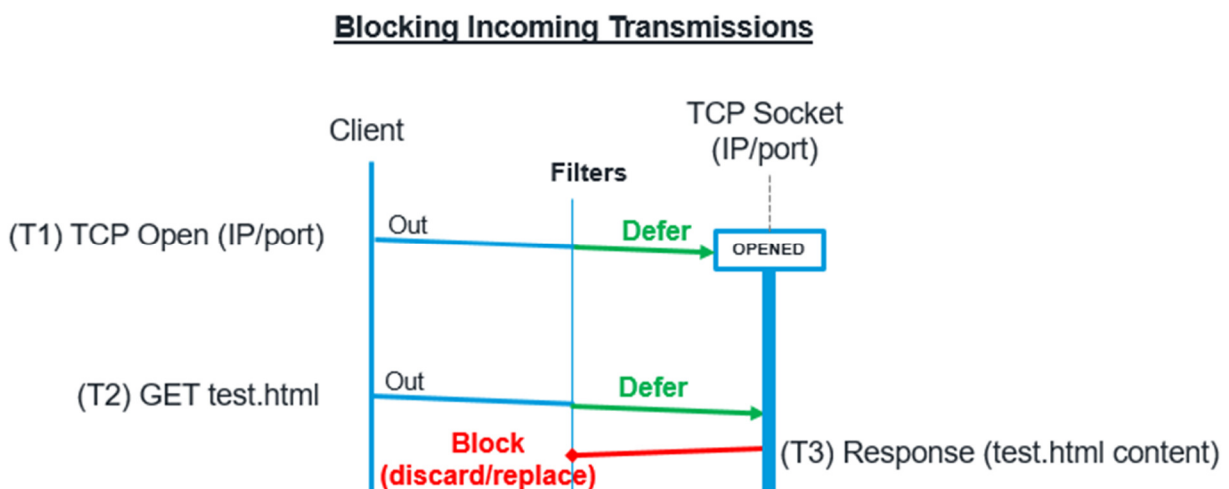
Instead of unconditionally censoring/allowing, users may want the ability to only block specific content known to be subjectively objectionable. *See id.*, 3:37-40. Duvall can block/allow specific pages or content from a domain after the content is received. *Id.*, 5:35-40. Duvall describes the same set of filters being applied to the outgoing transmission for that content. *Id.*, 5:35-40. The user also can target incoming content with the filters, allowing the system to censor in different ways (*i.e.*, “discarding incoming data ... or by replacing parts of received data”). *Id.*, 1:39-40. *See also id.*, 5:35-40.

Duvall thus provides additional actions that can instead “defer the decision whether to allow or block, and then monitor transmissions to search for a particular command and a particular filter pattern.” *Id.*, 1:46-48. This action therefore waits

to act on later transmissions in the TCP sequence. EX2001, ¶¶98. This type of action is a “deferred-action.”

Notably, the filtering process always remains the same, *i.e.*, Duvall just attempts to match the filtering criteria to the transmission in sequences until the first matching filter is found. But the action taken on each matched transmissions differs depending on the action specified and where the matched transmission is in the sequence of TCP transmissions (*i.e.*, T1, T2, or T3). *See* EX1004, 1:46-48; 6:19-26; *see also* EX2001, ¶¶99. Filters using deferred-actions can specify keywords and filter patterns that can be mapped directly to content to determine whether it is known to be objectionable and therefore should be censored. *See id.*, 5:8-19.

A visualization of the process described at EX1004, 6:10-28, can be seen below:



EX2001, ¶102.

In Duvall's example, incoming transmissions are responses to an outgoing request for data (*e.g.*, GET test.html, T2). *See* EX1004, 6:10-28; *see also id.*, 1:49-50. Outgoing transmissions (T1-T2) must be successfully transmitted before the host can respond with incoming data. EX2001, ¶103. Thus, Duvall can allow the data stream to open (T1) and permit the detected outgoing transmission (T2) while also queueing the filters for later incoming transmissions. *See* EX1004, 6:19-28, 5:30-35. Specifically, Duvall explains:

The filters are preferably queued so that filter pattern matching is performed when incoming data is received from the host server following a particular detected outgoing command. With such queueing, a similar scan of the message is performed on the outgoing transmission, but blocking is performed on the incoming message.

Id., 5:30-35.

An otherwise matching deferred-action filter will not have a chance to act if it is arranged after a matching direct-action filter because Duvall only acts according to the first match. EX2001, ¶¶106-105.

Thus, Duvall can defer action on the outgoing transmissions (T1-T2) by effectively just allowing them and waiting to match later incoming transmissions in the sequence. *Id.* When an incoming transmission is matched (T3), its data can be discarded or replaced. *Id.*

c. Duvall Acts on the First-Matched Filter

Duvall applies a single set of filters from its filtering database to each transmission and “takes action based on the first match[.]” EX1004, 4:30-31; *see also id.*, 1:42-52. Although Duvall categorizes the types of actions into two logical groups (*i.e.*, direct and deferred-action), each filter just exists as a filter entry in Duvall’s filtering database, regardless of its action type. *See id.*, 4:12-21. Duvall simply applies the same set of filters to each transmission when sequentially searching for the first singular match. *See id.*, 4:28-37, 5:8-29.

Importantly, if a deferred-action is desired, no other filters can take precedence when setting the ordering of the filters. EX2001, ¶¶105-106. Duvall also provides users with a default filter—which the user can use to BLOCK/ALLOW transmissions that do not match with another filter in the list. EX1004, 5:1-3. But data matched by the default filter is not data the user is uncertain about. Instead, it is data the user prefers to block or allow according to his subjective preference.

In short, Duvall’s entire system only filters based on criteria known to the user (*i.e.*, based on predetermination that certain content or source of content is objectionable). Duvall’s filters never identify anything that the user was uncertain about, which would rise to the level of a *potential* event. The user subjectively determines whether a transmission should be blocked or allowed through the

specification of its filters. Duvall therefore has no need for any second-level objective confirmation given this subjective nature.

3. Other Aspects of Duvall

Duvall discloses an updating mechanism allowing “existing filters in the filter database to be replaced with ... newly downloaded filters.” *Id.*, 7:27-29. While these updates are provided over the Internet, Duvall never discloses updating while operational. *See id.*, 7:18-29.

An editing manager “allows the user to edit the database” using “known database editing techniques.” *Id.*, 8:11-12. Users can “make custom changes if the user believes that the filtering system is filtering too much or too little.” *Id.*, 8:6-8. The editing manager allows users to “tweak” the overall system to censor more or censor less based on their subjective preferences. *See id.* But Duvall never suggests “filtering too much or too little” because a user is unsure whether something is objectionable.

Duvall contemplates editing the filters offline—not while the system is in operation. EX2001, ¶115. Duvall explains that “during a session over the Internet, a user can copy URLs for **later** editing.” EX1004, 8:13-14 (emphasis added). Then, after a session over the Internet, the user “can then copy those URLs for inclusion in the database, or can edit the entry.” *Id.*, 8:14-16.

The “filtering system can [also] be provided from a server 30 that is on the client’s own network 40.” *Id.*, 8:19-22. The “filter database [will be] located on server 30.” *Id.*, 8:26-27. An alternative search technique is needed to “be efficient enough for a server environment.” *Id.*, 8:39-42. Duvall therefore performs pre-processing on the server database to allow the filtering system to search more efficiently at runtime. *Id.*, 8:42-67. This pre-processing generates various tables that are used for searching. *Id.*, 8:39-61. This pre-processing is incompatible with dynamic updates because the tables must be generated whenever there is a change. EX2001, ¶117.

Finally, Duvall discloses patching code for its content filtering system. EX1004, 9:14-11:20. Duvall’s method for patching is designed to prevent end-users from bypassing Duvall’s censorship tool. *See id.*, 9:65-10:63; *see also id.*, 2:1-11.

B. Chu

Chu discloses a blacklist/whitelist filtering policy that filters URLs in the context of trust management. *See* EX1005 at 11, 43. The policy will return a response of either “*true*”, “*false*”, or “*unknown*,” indicating whether the URL associated with a file is trustworthy. *See id.*, 31-32; 43-44. Trustworthiness for Chu’s sample policy is based on the file’s source (*e.g.*, URL). *See id.*, 24; 43-44. If the source is trustworthy, then the information itself will be treated as

trustworthy, and vice versa. *See id.*, 24. When the policy returns “*unknown*” (*i.e.*, trustworthiness of the source cannot be determined) control is returned to the application seeking to access the file, and the end-user is prompted for attention by the application. *Id.*, 43-44. No definitive trust/do-not-trust determination is made, nor is any analysis conducted on the “*unknown*” results. Instead, “*unknown*” simply reflects the lack of any determination or analysis. *Id.*, 43-44.

1. Chu Focuses on Trust Management

Instead of focusing on analysis techniques to identify potential threats (*e.g.*, analyzing the content), trust management—according to Chu—considers the trustworthiness of third parties providing data. *See id.*; *see also id.*, 32. Chu does not determine whether a file is safe, only whether the party distributing the file is trustworthy. *See id.*, 11; *see also id.*, 32. Chu therefore explains that trust management is a “distinct aspect” of security. *Id.*, 11.

In defining trust management, Chu initially observes various concerns related to downloading a game from the web, including whether the game has a virus. *Id.*, 23. Instead of addressing these concerns by providing techniques to find viruses, Chu treats them as trust management issues. *Id.* In other words, for Chu, these concerns exist because the user cannot trust that the provider of the code is who the provider purports to be, especially in the Internet age. *See id.* 23-24. Chu lists three possible ways to address these concerns from a trust

management perspective. *See id.* The first approach limits code functionality using a sandbox so no trust or analysis is needed. *Id.* But “[n]o matter how much effort is put into building a robust sandbox, hackers can always find security holes to compromise the sandbox.” *Id.* The second approach requires additional code to prove the original code is safe. *Id.* This approach also is impractical because there are too many different types of executables, which all need different solutions. *Id.*

Ultimately, Chu prefers the third approach—code-signing. *See id.*, 24-25. Code-signing “uses cryptography to establish authentication and integrity of an untrusted piece of code.” *Id.*, 24. A third-party can “provide digitally signed metadata to express trust,” and applications can then decide whether to trust the code, based on whether they trust the third-party. *Id.* In other words, Chu looks to who the sender is. To that end, Chu teaches a “trust management infrastructure” facilitating this process. *See id.*, 25. Rather than identifying potential security risks by directly analyzing the data, Chu’s infrastructure determines whether any third-parties associated with the data are trustworthy.

2. Chu’s REFEREE Trust Management System

According to Chu, the “trust management infrastructure” has four key components: metadata format, trust protocol, trust policy language, and execution environment. *Id.*, 13. Together, they form Chu’s “trust management system.” *Id.* *Id.* According to Chu, its execution environment—REFEREE—“is the heart of a

trust management system.” *Id.*, 28. The various policies that Petitioner relies upon from Chu are disclosed in the context of REFEREE.

REFEREE uses a tri-valued logic operand, which has three possible outcomes: *true*, *false*, and *unknown*. *Id.*, 31-32. Importantly, *unknown* means that “REFEREE was unable to find sufficient credentials.” *Id.* When returned, “REFEREE forces the host application to decide what action (if any) should be taken.” *Id.*, 32. In other words, Chu merely indicates that it has failed to identify the source as trusted when *unknown* is returned.

3. Chu’s Sample REFEREE Policy

Chu discloses sample policies for REFEREE. *Id.*, 43-48. These policies “use code-signing as the target trust management problem.” *Id.*, 43. Therefore, Chu’s concern is about whether a URL corresponds to a trusted entity (*e.g.*, whether “**Error! Hyperlink reference not valid.**www.harverd.edu” is trusted), and not about identifying potential security risks. *See id.*, 44. One sample policy from Chu uses a whitelist and blacklist to filter URLs. *Id.* Chu observes that whitelist and blacklist filters require the user to know beforehand what sites should or should not be trusted. *Id.* Therefore, Chu explains that such a policy “is useful if the application knows *a priori* a list of sites or directories it should not download codes from.” *Id.*

Chu also explains that REFEREE will return “*unknown*” when the requested URL is in neither the blacklist nor the whitelist. *Id.* And when “*unknown*” is returned by REFEREE, the application will “prompt [the end-user] for [her] attention.” *Id.*, 44. The “blacklist and whitelist ensure good automation of the trust decision process,” because the end-user need not be prompted to intervene so long as the URL is in either list. *Id.* “User intervention is needed only when the given URL is in neither the blacklist nor the whitelist.” *Id.* End-users are left to choose whether or not to accept the risk of proceeding with code from an unknown source. But that choice has no bearing on the (future) trustworthiness of the URL or determining whether the data/URL itself is actually problematic. Nor does it suggest the end-user could perform any further analysis to make such a determination.

C. Cogger

Cogger discloses an interface for customers to remotely access a telecommunications service provider’s pre-existing trouble ticketing system. In “conventional customer enabled maintenance systems, a connection is made with a large legacy system via a dial-up connection from a customer owned personal computer.” EX1033, 1:26-28. Notably, “dial-up access requires custom software on the customer workstation to provide dial-up services, communication services, emulation and/or translation services and generally some resident custom form of

the legacy application to interface with the mid-range or mainframe computer running the legacy system.” *Id.*, 1:32-36. Therefore, it is desirable to “provide connectivity to enterprise legacy systems over the public Internet, as the Internet provides access connectivity [to the] world wide [web] via the TCP/IP protocol, without need to navigate various telephone exchanges, dialing standards or signal standards.” *Id.*, 2:1-5.

Cogger focuses on a “Service Inquiry” (“SI”) application, which is part of a suite of “telecommunications network applications” allowing customers to “manage their telecommunication assets, quickly and securely.” *Id.*, 13:66-14:4; 4:65-5:1. The suite of applications are interoperable and “leverage existing infrastructure services.” *See id.*, 13:66-14:17.

With respect to SI, Cogger explains that trouble tickets were already being used in the telecommunications industry’s fault management systems. *Id.*, 2:6-11. Trouble tickets “represent[] the set of ongoing and future repair efforts of the service provider.” *Id.*, 2:26-28. But customers want “an open access route to this information” and “to access their data over the internet.” *Id.*, 2:44-47. Cogger therefore provides a “system and method for allowing a customer to remotely access a service provider’s trouble ticketing system.” *Id.*, 2:39-43.

For SI, Cogger discloses an architecture including a client (Fig. 2, 20), a Demilitarized Zone (DMZ) (Fig. 2, 17), a MCI Intranet Dispatcher Server (Fig. 2,

26), MCI Intranet applications servers (Fig. 2, 30), and data warehouses and legacy systems (Fig. 2, 40). *Id.*, 6:8-14.

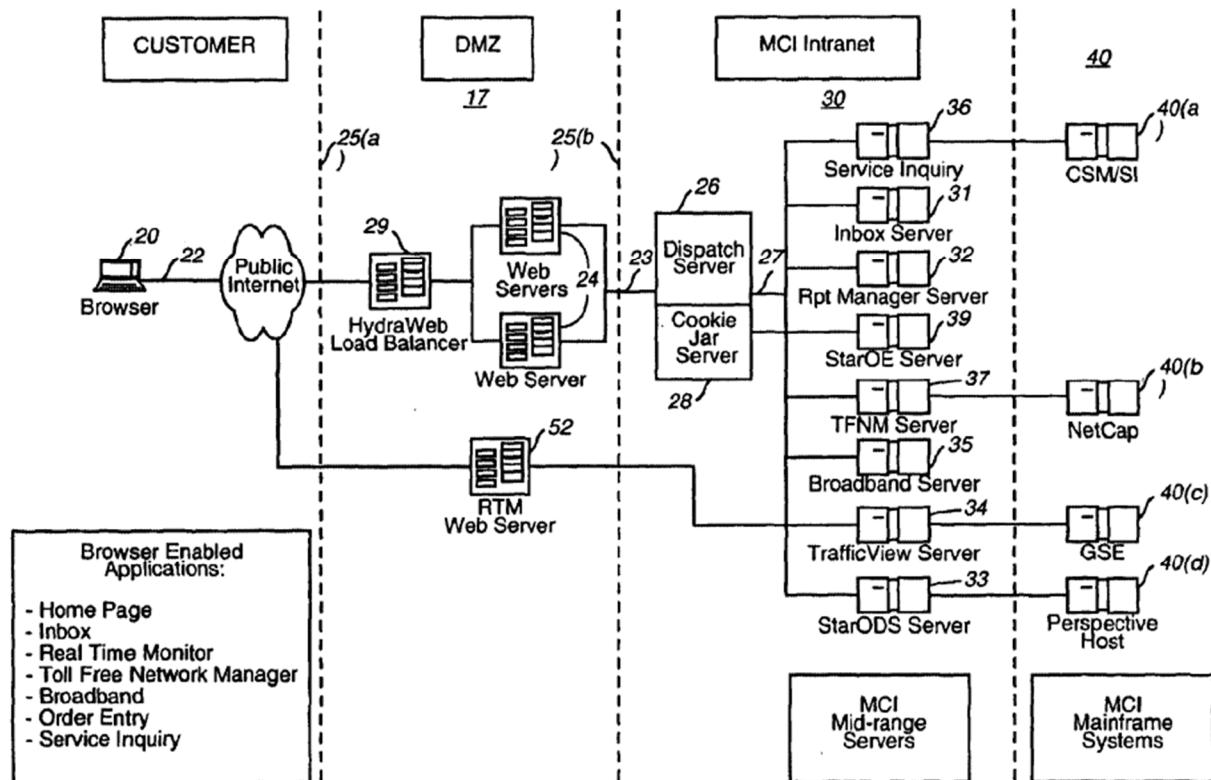


FIG. 2

Id., Fig. 2.

This architecture translates SI requests into requests processable by the legacy trouble ticketing system—*i.e.*, the Customer Service Management System (CSM). *Id.*, 15:58-16:31. Translation relies on various components within the architecture and the existing suite of telecommunications applications. *Id.*, 15:58-16:31. Once translated, an actual trouble request can be issued to the legacy CSM. *Id.*, 16:28-30. The legacy CSM processes the request and returns the response

data—passing it back to SI. *Id.*, 16:32-42. Because the legacy system already exists, Cogger does not actually disclose the legacy CSM’s features and functionality in detail.

Instead, Cogger only discloses the front-end interface for SI—explaining how it can be used to interact with legacy CSM. For example, a user may click a “Create” button to create a new ticket in SI. *Id.*, 21:55-57. Because the existing legacy CSM system typically requires an “Event-Type/Call-Type” for most transactions, SI provides the interface that “asks the customer the problem/identifier type and then narrow[s] down the problem by having the customer choose from a list of Product types, Service types and Trouble Descriptions.” *Id.*, 21:23-31. These data elements are passed on to the CSM legacy system to form the appropriate CSM request to open a Trouble Ticket. *See id.*, 21:41-57.

V. LEVEL OF SKILL IN THE ART

The level of skill in the art is immaterial to the PTAB’s decision on whether to institute *inter partes* review. Patent Owner therefore does not currently dispute Petitioner’s characterization but reserves the right in the future for this and/or other proceedings.

VI. CLAIM CONSTRUCTION

The words of the claim are given their ordinary and customary meaning as understood by persons of ordinary skill in the art at the time of the invention.

Phillips v. AWH Corp., 415 F.3d 1303, 1312-13 (Fed. Cir. 2005) (en banc).

Mindful that the Board will only “construe terms ‘that are in controversy and only to the extent required to resolve the controversy,’” Patent Owner submits that “post-filtering residue” has been misconstrued and misapplied by Petitioner. *Nidec Motor Corp. Zhongshan Broad Ocean Motor Co.*, 868 F.3d 1013, 1017 (Fed. Cir. 2017) (quoting *Vivid Techs., Inc. v. Am. Sci. & Eng’g, Inc.*, 200 F.3d 795, 803 (Fed. Cir. 1999)).

A. “Post-Filtering Residue”

Certain limitations in the Schneier patent were previously construed during a litigation in District Court against Fortinet. *See British Telecommunications plc v. Fortinet, Inc.*, No. 1:2018cv01018 (D. Del. 2019). There “post-filtering residue, wherein the postfiltering residue is data neither discarded nor selected by filtering” was construed to mean “status data that undergoes negative and positive filtering, but is neither discarded by such negative filtering nor selected by such positive filtering.” EX1013 at 4. Petitioner acknowledges this and suggests “the challenged claims are unpatentable under either the ordinary and customary meaning ... or the district court’s previous claim constructions.” Petition, 12-13.

But Petitioner never explains why the District Court’s claim construction diverges from the ordinary and customary meaning of the term. Indeed, Petitioner never provides its view of the ordinary and customary meaning of the term. To the contrary, Petitioner contends no construction is needed. Petition. at 12

Despite apparently arguing for no construction, Petitioner nonetheless offers an implicit construction, suggesting that the “plain language of this element does not require any data to actually be ‘discarded’ or ‘selected by filtering’—only that the ‘post-filtering residue is data neither discarded nor selected by filtering.’” *Id.* But this implicit construction is flawed. Petitioner omits the requirement that the “residue” be “post-filtering” and impermissibly broadens the residue to include any data—not just status data.

First, the limitation’s plain language requires the residue to be “post-filtering,” which means the residue produced after filtering is complete. *See* EX1001, claims 1, 18. Indeed, this is consistent with the rest of the language requiring that the “analysis includes filtering followed by an analysis of post-filtering residue.” *Id.* Clearly, the claims not only require the filtering process to produce the residue, but that it be “followed by” a separate analysis performed on the **post**-filtering residue. *Id.* Only if the filtering process is completed can it be “followed by” analysis of the post-filtering data. *Id.* Indeed, Schneier’s filtering subsystem 2030 is distinct from the anomaly engine 2050 that performs the

required analysis. *See id.*, 8:55-62. Therefore, under the proper construction, the filtering process must be distinct from the analysis that follows the filtering.

Petitioner ignores this distinction and suggests in-flight data—that is still in the process of being filtered (*i.e.*, mid-filtering data)—would qualify as “post-filtering residue.” *See* Petition, 16. Mid-filtering data clearly is not “post-filtering” residue because filtering has yet to complete. Petitioner cannot rewrite the claims so that “post-filtering” means “mid-filtering.” Nor can Petitioner ignore the terms “followed by” or “residue.” *See Stumbo v. Eastman Outdoors, Inc.*, 508 F.3d 1358, 1362 (Fed. Cir. 2007) (denouncing claim constructions that render phrases in claims superfluous), *see also Bicon Inc. v. Straumann Co.*, 441 F.3d 945, 950 (Fed. Cir. 2006) (“claims are interpreted with an eye toward giving effect to all terms in the claim”).

Second, Petitioner ignores that “post-filtering residue” is residual status data, not just any residual data. Indeed, the rest of the limitation recites “analyzing status data ... wherein the analysis includes filtering followed by an analysis of post-filtering residue.” *E.g.*, EX1001, claim 1. Because the analysis of post-filtering residue is specifically to analyze status data, the “post-filtering residue” must refer to post-filtering residue status data. Indeed, this is consistent with the Fortinet District Court’s prior construction. *See* EX1013 at 4 (construing “post-filtering residue...” to mean “status data that undergoes negative and positive

filtering, but is neither discarded by such negative filtering nor selected by such positive filtering”).

Petitioner glosses over this detail when analyzing the claims. Petition, 31-32. Though IP addresses are status data, Petitioner bears the burden to show how Duvall actually analyzes any leftover status data to identify potentially security-related events. Denial of the petition is warranted because Petitioner never addresses this.

**VII. THE PETITION FAILS TO ESTABLISH A REASONABLE
LIKELIHOOD THAT THE ASSERTED REFERENCES RENDER
ANY CLAIM UNPATENTABLE**

A. Petitioner Fails to Meet Its Burden

Petitioner bears the burden of demonstrating, with particularity, why there is a reasonable likelihood a claim of the patent will be found unpatentable. *See Harmonic Inc. v. Avid Tech., Inc.*, 815 F.3d 1356, 1363 (Fed. Cir. 2016). This burden never shifts to Patent Owner. *Dynamic Drinkware, LLC v. Nat’l Graphics, Inc.*, 800 F.3d 1375, 1378 (Fed. Cir. 2015). For obviousness Petitioner also has the “burden to prove that all claimed limitations are disclosed in the prior art.” *See PAR Pharm., Inc. et al. v. Twi Pharms, Inc.*, 773 F.3d 1186, 1194 (Fed. Cir. 2014); *Medichem, S.A. v. Rolabo, S.L.*, 437 F.3d 1157, 1164 (Fed. Cir. 2006) (confirming that courts consider motivation to combine and reasonable expectation of success only “if all the elements of an invention are found in a combination of references”).

Even if all the limitations are found (not the case here), Petitioner must demonstrate why a skilled artisan would be motivated to combine the references.

Polaris Indus. v. Artic Cat, Inc., 882 F.3d 1056, 1068-69 (Fed. Cir. 2018).

Moreover, it is impermissible to use *Schneier*, in hindsight, as a template for arbitrarily picking and choosing limitations from the prior art to combine them to arrive at the claimed invention. *In re NTP, Inc.*, 654 F.3d 1279, 1299 (Fed. Cir. 2011) (explaining the patent-at-issue may not be used “as a guide through the maze of prior art references, combining the right references in the right way so as to achieve the result of the claims in suit”) (citations omitted).

As detailed below, Petitioner’s references do not disclose multiple claim limitations. Petitioner also fails its burden to demonstrate why a POSA would combine the cited references with a reasonable expectation of success.

B. Multiple Features Present in Every Claim Are Missing from Petitioner’s Asserted References

The claims of *Schneier* contemplate two separate levels of analysis of status data after that status data has been positively and negatively filtered. These two separate levels of analysis help determine whether unknown and new threats are reflected in the status data that did not match positive and negative filters. In contrast, *Duvall* and *Chu* fail to teach even one level of analysis after filtering, let alone the two-level process required by the claims (first at the probe to identify

potential events, then transmitting to an analyst system, *e.g.*, at a SOC for a confirmation step). In addition, both references are also completely silent as to any dynamic update capability.

1. Duvall and Chu Lack Analysis of Post-Filter Residue Status Data to Identify Potential Security-Related Events

Neither Duvall nor Chu performs a first analysis, after filtering, on post-filtering residue, to flag potential events, at the probe, before transmission on for more detailed analysis. EX2001, ¶¶141-155. This first-level of analysis to flag potential events takes place at the probe. EX1001, 3:17-21; EX2001, ¶142. The claim language appears in element b of claim 1 as follows:

analyze status data to identify potentially security-related events represented in the status data, wherein the analysis includes filtering followed by an analysis of post-filtering residue, wherein the post-filtering residue is data neither discarded nor selected by filtering[.]

EX1001, 33:30-35 (claim 1) (emphasis added); EX2001, ¶143.

It can also be found within the first step of independent method claim 18 as follows:

creating an event record for information received about an identified potentially security-related event occurring on the network, wherein the potentially security-related event is identified by filtering followed by an analysis of post-filtering residue, wherein the post-filtering residue is neither discarded nor selected by the filtering[.]

EX1001, 34:25-30 (emphasis added).²

The independent claims of Schneier all require the same thing. After filtering, the post-filtering residue status data, which was neither discarded nor selected (by positive and negative filtering), is analyzed, at the probe, to identify a potentially security-related event. Neither reference cited by Petitioner provides even the slightest suggestion to do that. Instead, both Duvall and Chu only teach filtering for things that are already known. Neither of them analyzes any data to identify *potential* threats after filtering (*i.e.*, previously unknown threats from the post-filtering residue).

Petitioner argues that Duvall shows a first-level of analysis of post-filtering residue. *See* Petition 30-38. Petitioner is wrong. Duvall merely teaches a filtering system for censoring subjectively objectionable material. EX2001, ¶¶146-148. Petitioner improperly characterizes data still being filtered (*i.e.*, mid-filtering data) as post-filtering residue. And Petitioner strains its reading of Duvall to argue that portions of Duvall’s subjective-criteria based filtering (*i.e.*, whether personally

² The Court in *Fortinet* construed “information received about an identified event ... wherein the potentially security-related event is identified by filtering” to mean “information received **from a probe** about an identified event ... wherein the potentially security-related event is identified **at the probe** by filtering.” EX1013, 2 (emphasis added).

objectionable) somehow equates to an objective analysis that identifies potential security-related events. Moreover, Petitioner completely ignores that this first-level analysis must be of status data—not just any data involved in the filtering process. *See, supra*, VI.A.

First, Duvall never “identifies potentially security-related events.”

Petitioner improperly equates Duvall’s subjective-criteria based filtering, which only matches to content known to be subjectively objectionable, to an analysis that identifies events that are potentially objectively malicious. *See* Petition 33-36.

Duvall uses filters to censor data based on subjective preferences; it never analyzes any data for potentially security-related events. EX2001, ¶146. Whether an identified event is in-fact potentially security-related is an objective question relevant to network security. Objective analysis is needed to elevate the data not matching any known filtering criteria to “potentially” security-related. But such an objective determination makes no sense for Duvall where the question of censorship is purely subjective. *Id.*, ¶147.

Duvall is limited to filtering out the known objectionable/non-objectionable material, which is determined based on user preferences. *Id.*, ¶147-148. The status of Duvall’s data (including data “neither ALLOWED nor BLOCKED”) is always known to the user because the user subjectively pre-determines whether it is objectionable. *Id.* Duvall never contemplates events that are unknown and such

that an objective analysis could elevate them into “potential” events at a first-level or be confirmed at a second-level. No such analysis is ever needed by Duvall, and no such analysis is ever taught.

Conceding Duvall is not security-related art, Petitioner instead argues that certain filtered content “can carry viruses or malware.” Petition, 24. But that still does not mean Duvall teaches or even suggests “**identifying**” any “potential security-related events.” EX1001, claims 1, 18. Even if a security-related event could tag along with subjectively objectionable content, filtering the content would not amount to identifying the security-related events. Indeed, security-related events could tag along with virtually any content—even non-objectionable content. EX2001, ¶150.

Petitioner also argues that it would be obvious to use Duvall for security to block malware. Petition, 25. But Duvall’s filtering system is unsuitable to monitor network security. *See, infra*, VII.C.1 (*e.g.*, lack of objective analysis, limited visibility into threats, lack of concern for network security). Petitioner provides no explanation why a POSA would use Duvall for security despite that unsuitability.

Second, Duvall does not analyze post-filtering residue. Partially filtered data that is still being processed by a common and pre-defined sequence of filters (*i.e.*, mid-filtering data) cannot be “post-filtering” residue (*i.e.*, after filtering) under any reasonable construction of that term, let alone the construction adopted by the

District Court in *Fortinet*, which should also apply here. Duvall discloses a single filtering process that is always applied in a pre-defined sequence, and arbitrarily dividing that sequence mid-filtering does not somehow create the required distinct positive filtering, negative filtering, and post-filtering analysis steps. Petitioner's characterization to the contrary is incorrect—and indeed inconsistent with how Duvall characterizes itself. *Compare* Petition, 16-18, 23-26, 30-34, *with, e.g.*, EX1004, 4:22-37. Indeed, it is improper to use Schneier “as a guide ... so as to achieve the result of the claim[.]” *In re NTP*, 654 F.3d at 1299.

Duvall applies a single set of filters from its filtering database to each data transmission—acting based on the first match. *See* EX1004, 4:22-37. Duvall suggests efficiently ordering filters with direct-actions first, followed by those with deferred-actions. But filter order is dictated by the desired censorship. EX2001, ¶153. Therefore, Petitioner's characterization of Duvall's deferred-action filter as analysis of post-filtering residue is strained because it only applies when the filters happen to be ordered a specific way—and even then, Duvall is still in the process of searching for a matching filter, meaning filtering is not complete. *See* EX1004, 4:65-5:7. Any mid-filtering data still being considered is not yet “post-filtering” data, much less analysis of post-filtering residue status data.

Petitioner mischaracterizes the deferred-action filter's filtering criteria as “techniques for analyzing residue data.” Petition, 18. (citing EX1004, Fig. 4).

Petitioner is wrong. As explained above, the data passing the direct-action filters in Duvall is not unknown residue to be analyzed. *See supra*, 39. And the latter filters are just finishing Duvall's process for filtering what is already known.

Therefore, Petitioner has failed to carry its burden of demonstrating, with particularity, that the references disclose the limitation specifically identified by the Examiner at the time of allowance—filtering status data into residue, followed by analysis of the post-filtering residue to identify a potentially security-related event. That limitation is present in every claim. Institution should be denied for at least this reason. *Medichem, S.A.*, 437 F.3d at 1164.

2. Duvall and Chu Lack a Second-level of Analysis By an Analyst System

Duvall and Chu also fail to disclose the second-level of analysis with an analyst system of potential events identified in the first-level of analysis. Having a first-level analysis to identify potential threats is a prerequisite for the second-level of analysis, which confirms those threats. EX2001, ¶156. The relevant language is in, element c of claim 1:

transmit information about the identified events to an analyst system associated with the security monitoring system[.]

EX1001, 33:36-38 (claim 1) (emphasis added); EX2001, ¶157.

Again, neither asserted reference provides even the slightest suggestion to transmit information about previously identified events from a first-level of

analysis to an analyst system that can determine whether the potential events are actual events and/or resolve them. *Id.* To the extent there is any sort of analysis taught by the references (which there is not), analysis could only relate to the first-level analysis—and not the second. Duvall is silent as to any second-level of analysis. *Id.*, ¶159. Indeed, Petitioner does not even attempt to argue Duvall satisfies this limitation. And nowhere does Petitioner credibly explain why a POSA would want to add a second-level analysis to Duvall (*e.g.*, for confirmation), when Duvall only identifies what is already known (*i.e.*, no confirmation is needed).

Petitioner relies on Chu to teach a second level of analysis (Petition, 37-40), but, like Duvall, Chu never discloses any second-level analysis. Indeed, Chu cannot teach “transmitting information about **said identified events**” from a first-level analysis when there was no first-level analysis to begin with. EX1001, 33:36-38 (claim 1).

Regardless, Chu does not “transmit[]” any information to an analyst system. Nor does Chu examine any information about potential events (*e.g.*, for a confirmatory step) that would suggest transmitting that information to an analyst system. Chu only filters URLs with whitelists and blacklists to automate a trust decision process. *See* EX1005, 44. If the trust decision cannot be automated because the URL is neither in the whitelist nor the blacklist, then Chu simply

returns control back to the end-user—alerting them that no decision could be made.

Id. Chu characterizes this as “user intervention” because the system cannot automate the decision here—instead, giving control back to the end-user. *See id.*; EX2001, ¶161. But Petitioner’s attempt to argue that “user intervention” is the same as transmitting information about said identified events to an analyst system fails. *See* Petition 21-23; 37-40. Deferring control back to the application due to an inability to make a trust decision is different than transmitting information about identified potentially security-related events to an analyst system (*e.g.*, for a confirmatory analysis). *See, e.g.*, EX1001, 2:43-64. Nor would it make sense to redirect the alert away from the end-user’s application to a (non-existent) analyst system for any reason.

Neither Chu nor Duvall teaches or suggests (a) any previously identified potential events, (b) “transmitting information about” any such events, or (c) using an “analyst system.” Each missing limitation is addressed in turn.

a. Chu and Duvall Never Teach or Suggest Any Previously “Identified Events” from a First-level Analysis

Schneier recites “transmitting information about said identified events.” EX1001, 33:36-38 (emphasis added). Because Duvall and Chu never teach the first-level analysis to identify potential events, there is no information on potential events to transmit. As discussed, VII.B.1, Duvall does not disclose the first-level

analysis. And Chu similarly does not identify potentially security-related events. Rather Chu only determines whether a file is trustworthy based on known criteria. When REFEREE returns “*unknown*,” Chu merely informs the end-user that REFEREE’s analysis was inconclusive. *See* EX1005, 43-44. But informing the user that an analysis was inconclusive neither teaches nor suggests any further analysis. And as Chu itself explains, its trustworthiness determination is not the same as identifying a potentially security-related event or even identifying an actual security event. *Id.*, 11. Moreover, Petitioner effectively treats the data that triggered the “*unknown*” response by Chu as an identified potential event. But that makes little sense because Chu has not analyzed that “*unknown*” data to elevate it from just “unknown” into a “potential” security-related event.

b. Chu Never “Transmit[s] Information” After Filtering Besides REFEREE’s “*unknown*” Response

Even if the “*unknown*” data (*i.e.*, data triggering REFEREE’s response of “*unknown*”) somehow was associated with an identified potential security event (which it is not), Chu’s “user intervention” still would not involve transmitting any information about that event to anyone. *See* Petition 22-24; 38-39. Petitioner mischaracterizes Chu’s “user intervention” as somehow teaching the transmission of data to a human for a confirmatory analysis. *See id.*, 40. But at best, Chu’s sample policy only returns the response “*unknown*” and defers control back to the

original application, which just alerts the end-user of the lack of any determination. *See* EX1005, 43-44. REFEREE never transmits the “*unknown*” data itself to a human, *id.*, nor would it make sense to, given the end-user already has that data, *see* EX2001, ¶167. Therefore, Chu’s “user intervention” alert does not even transmit information about the “*unknown*” data to a human—let alone information about previously identified potential events. *Id.*, ¶168.

c. Neither Duvall Nor Chu Discusses an Analyst System to Which Information Could Be Transmitted

At best, Chu’s “user intervention” teaches prompting the end-user for attention at the original application. *See* EX1005, 44. This merely lets the end-user decide whether to access the file. EX2001, ¶169. No analyst system is involved whatsoever. The end-user’s application, which includes the probe, cannot be the claimed analyst system. *Id.*, ¶169. Indeed, nowhere does Chu mention using an analyst system. Duvall never mentions an analyst system either.

Apparently recognizing the lack of an analyst system (and the unsuitability of returning control to the original application in Duvall, *see infra*, pp. 64-65), Petitioner suggests that a trained network administrator from a specialized third-party network-security service could serve as an analyst. *See* Petition, 22. In support, Petitioner cites to a reference not actually relied upon in the proposed grounds for unpatentability. *See* Petition, 22 (citing EX1037). But neither the

Petition nor this third reference discloses how to introduce the analyst system into either Duvall or Chu, much less *why* it would have been obvious to (1) shift the prompts away from the end-user to the administrator, (2) modify those alerts to include information about identified potential security-related events rather than returning “*unknown*”, or (3) have the administrator that handles filters do double-duty to analyze security-related events. Petitioner simply makes this up using Schneier as a template. Similarly, the Petition never explains how these modifications would be made, which combinations would yield a predictable result, or how the modified system would even operate. Obviousness cannot be reached absent these explanations. *See ActiveVideo Networks, Inc. v. Verizon Commc'ns, Inc.*, 694 F.3d 1312, 1327 (Fed. Cir. 2012).

3. The Limitations Regarding Receipt of Feedback at the Probe to Dynamically Enhance Its Analytical Capabilities Are Missing from the Asserted References

Independent claim 1 of the ‘641 Patent contains a limitation directed to updating the analytical capabilities of the probe, dynamically, through the receipt of feedback at the probe. EX2001, ¶171. The relevant claim language appears in elements d and e of claim 1 as follows:

d) a receiver for receiving feedback at the probe based on empirically-derived information reflecting operation of the security monitoring system; and

e) a modification control system for dynamically modifying an analysis capability of the probe during operation thereof based on the received feedback.

EX1001, 33:39-45; *see* EX2001, ¶¶171.

Duvall and Chu both fail to teach this dynamic feedback loop. Importantly, the term “dynamically” was previously construed to mean “during actual operation, rather than offline.” EX1013, 2. Petitioner does not dispute this construction. Instead, Petitioner wrongly argues that Duvall teaches a dynamic feedback loop. *See id.* 41-43. Petitioner’s argument is flawed because Duvall never discloses “dynamically” modifying any analysis capability, and Duvall’s algorithm for its server-based system is actually incompatible with “dynamic” modification. Critically, any modifications to Duvall’s filters are based on subjective views; not on “feedback at the probe based on **empirically-derived information** reflecting operation of said security monitoring system.”

First, Petitioner speculates that a POSA would have recognized that the analysis capability of Duvall is modified dynamically “because taking the system offline each time an update was required would be disadvantageous.” Petition, 41. But just because taking the system offline would be disadvantageous does not mean Duvall teaches modifying its analysis capability “during actual operation, rather than offline.” *See* EX1013, 2. Petitioner argues that Duvall “avoid[s] this disadvantage” by “provid[ing] updates online so that the system can adapt as new

sites and servers are added to the Internet.” *Id.* (citing EX1004, 2:16-18). But the word “online” in Duvall means “over the Internet”—not while the system is operational. EX2001, ¶174; *see* EX1004, 2:16-18; *see also id.*, 7:17-29 (explaining that “[b]ecause Internet sites are being added to the Internet at a fast rate” an update mechanism is provided to “locate a particular ... update server [] over the Internet”). Merely updating over the Internet (*i.e.*, online) neither teaches nor suggests “dynamically” modifying an analysis capability “during actual operation, rather than offline.” Compare, *id.*, 7:17-29, with, EX1013 at 2.

Second, Petitioner argues Duvall’s filtering system accommodates dynamic updates because it uses a database. Petitioner appears to be arguing that a database inherently allows for dynamic updates—which is false. *See* [Lee Decl.]. In fact, Duvall actually teaches the opposite. *See* EX1004, 8:12-14 (“during a session over the Internet, a user can copy URLs for **later editing**.” (emphasis added)).

Petitioner provides no evidence besides its expert’s conclusory statements. *See* EX1003, ¶151. This bare conclusion carries no weight. *See* 37 C.F.R. § 42.65(a) (“expert testimony that does not disclose the underlying facts or data on which the opinion is based is entitled to little or no weight”); *see also In re Magnum Oil Tools Int’l, Ltd.*, 829 F.3d 1364, 1380 (Fed. Cir. 2016). (“To satisfy its burden of proving obviousness, a petitioner cannot employ mere conclusory statements.

Petitioner must instead articulate specific reasoning, based on evidence of record,

to support the legal conclusion of obviousness.”); *ActiveVideo Networks, Inc. v. Verizon Commc’ns, Inc.*, 694 F.3d 1312, 1327 (Fed. Cir. 2012) (affirming finding of non-obviousness when expert’s testimony “was conclusory and factually unsupported”).

Moreover, Duvall teaches pre-processing the database to “substantially reduce” the number of comparisons made when searching the database during operation. *Id.*, 8:40-42, 9:9-11. This requires “analy[sis] to determine the frequency of combinations of characters” and the creation of two additional tables for searching. *Id.*, 8:39-61. This is incompatible with “dynamic” modification because the two tables need to be generated whenever the filters are modified before searching is possible. EX2001, ¶¶176-177. The filtering system would be inoperable while the database performs the preprocessing. *Id.* Therefore, Duvall’s server-based system teaches against “dynamic” modification.

Third, Duvall’s filters are modified based on the user’s “belie[f] that the filtering system is filtering too much or too little.” EX1004, 8:7. But subjective beliefs and the user’s “individual tastes and sensitivities” are not objectively verifiable. *See id.*, 2:12-16. Duvall’s feedback is therefore based on the exact opposite of “empirically-derived information.” In fact, the subjective nature of Duvall precludes the empirically-driven approach disclosed by Schneier. EX2001, ¶ 178.

Elements d and e of every claim of the ‘641 Patent are missing from the asserted references. Chu is silent about dynamic modification of filters. *Id.*, ¶¶120, 165. Consequently, neither reference shows the limitation present in every claim directed to updating an analysis capability of the probe, dynamically. For this reason, alone, the Petition fails as a matter of law. *Medichem, S.A.*, 437 F.3d at 1164.

C. There Is No Basis for Combining Duvall and Chu, No Details How to Do So, and No Reasonable Expectation of Success

Petitioner bore the burden of showing that a skilled artisan not only “*could have made* but *would have been motivated to make* the combinations or modifications of prior art to arrive at the claimed invention.” *Personal Web Tech. v. Apple Inc.*, 848 F.3d 987, 993-94 (Fed. Cir. 2017) (citing *Belden v. Berk-Tek LLC*, 805 F.3d 1064, 1073 (Fed. Cir. 2015)) (emphasis in original). Petitioner had to show both how and why a POSA would have been motivated to combine Duvall and Chu. *TriVascular, Inc. v. Samuels*, 812 F.3d 1056, 1066 (Fed. Cir. 2016). Particularly, the Federal Circuit has held that *prima facie* obviousness requires the challenger (1) “to explain how specific references could be combined”; (2) “which combination(s) of elements in specific references would yield a predictable result”; and (3) “how many specific combinations would operate or read on” the claims. *ActiveVideo*, 694 F.3d at 1327-28. The Board has repeatedly recognized those

requirements and rejected obviousness allegations that fail to explain how the purported prior art elements could have been combined to achieve predictable results and arrive at the claimed invention. It was also incumbent upon Petitioner to avoid using the invention, retrospectively, to stitch together the prior art, with the benefit of hindsight. *Metalcraft of Mayville, Inc. v. The Toro Co.*, 848 F.3d 1358, 1367 (Fed. Cir. 2017).

There is no reason a POSA would be motivated to combine Duvall and Chu to arrive at anything beginning to approach the claimed invention. Not only are Petitioner's suggested motivations expressly contradicted by the disclosures of Duvall and Chu, but Chu does not even teach the features (*i.e.*, techniques for determining whether to block or allow residue) Petitioner attempts to combine into Duvall.

Duvall and Chu have completely different focuses: They attempt to solve different problems in different ways compared to each other and Schneier. Duvall focuses on censorship, whereas Chu focuses on trust management. *Compare*, EX1004, 1:10-29, *with*, EX1005, 11-12. The mere fact that Duvall and Chu both use filters is insufficient to show why or how a POSA would have combined their teachings, much less in a manner involving a network security approach as claimed. Indeed, Petitioner never explains why a POSA would combine a censorship mechanism with a trust management system to create a security

monitoring device. EX2001, ¶181. Duvall and Chu filter different things for different purposes in different ways from both each other, and from Schneier's claims. *Id.*

Accordingly, Petitioner's proposed combination is flawed for the many reasons discussed below.

1. Duvall Is Unsuitable for Use as a Network Security Monitoring System

Petitioner initially argues that Duvall recognizes its own system could “filter too much or too little.” Petition, 19-22, 37, 39-40. Petitioner's expert argues that a desire to “ensure that the system correctly filters data on a consistent basis” would provide the motivation to “seek additional techniques to more accurately resolve the status of residue status data transmissions to ensure transmissions are correctly blocked or allowed.” EX1003, ¶103. But this already presumes that Duvall would be used for security in the first place, which it is not. For security monitoring systems, like that in the '641 Patent, the goal is to detect and respond to network threats. *See* EX1001, 1:18-20. Schneier detects threats using objective analyses and improves using empirically-derived information, which is fundamentally unlike determining whether something is subjectively objectionable. EX2001, ¶183.

Duvall is only concerned with censorship. *Id.*, ¶184. Indeed, Duvall’s user is not uncertain about the status of any data matched by the default filter such that further confirmatory analysis would be needed. *See* EX1004, 5:1-7. Instead, Duvall intentionally relies on the default filter to allow or block transmissions (*e.g.*, where users want to block access to all but a few sites). *See id.*, 7:7-8. It is nonsensical to seek objective techniques to resolve the status of the data when what matters is subjective in nature and is pre-determined in advance. EX2001, ¶184. Therefore, Petitioner’s proffered desire is nonsensical.

a. Duvall Is Unsuitable for Network Security Because It Is Predicated on Subjective Preferences Predetermined by the User

Petitioner’s proffered desire to “seek additional techniques to more accurately resolve the status of residue” actually suggests Duvall should not be used for security in the first place. Security requires analysis that uses objective criteria (*i.e.*, is it a threat?)—which Duvall lacks. Duvall never teaches or discloses “ensur[ing] content that implicates security concerns does not reach recipients.” *Compare* Petition, 13, *with*, EX1004. Duvall never mentions security at all. Far from it, Duvall explicitly teaches that its filtering system filters content that is “personally objectionable” based on the user’s subjective “tastes and sensitivities.” *Id.*, 2:12-16.

Petitioner never explains why Duvall should be altered for use in such an unintended way. Petitioner asserts a “POSA would have recognized that data transmissions from unknown or questionable sources can carry viruses or malware.” *Id.*, 24. But that still does not explain why Duvall should be transformed to detect and block those viruses instead of the objectionable content. Moreover, a POSA would also have recognized that data from known and trustworthy sources could—likewise—carry viruses or malware. EX2001, ¶186.

Duvall notes in passing that its “implementing software ... can [] be incorporated into a firewall,” but never suggests it would become a security monitoring system. EX1004, 1:61-63. Instead, Duvall just acknowledges it could be convenient to enforce the censorship from a central point through which internet traffic flows—the firewall. EX2001, ¶187.

Because Duvall is not focused on security, Duvall also never contemplates an objective analysis that could identify potential security-related events. Not only is Petitioner’s asserted motivation contrary to Duvall’s explicit disclosure, but this supposed desire to seek additional techniques only arises because Petitioner is attempting to transform Duvall into something Duvall itself never intended.

Petitioner suggests a POSA would be motivated to look for Chu to add a confirmatory step to Duvall after fundamentally transforming the nature of Duvall into a security device that needs Chu. This reasoning is circular and clearly driven

by hindsight. Petitioner's asserted motivation highlights one of the many reasons why Duvall's filtering system is unsuitable as a security monitoring device—because it lacks any consideration for objective analysis.

b. Duvall Is Unsuitable for Computer Network Security Because It Is Not Concerned with Security Threats

Duvall demonstrates that a censorship-based system operates fundamentally differently than a security-focused system. For example, Duvall only suggests monitoring transmissions related to a particular type of transport protocol commonly used to receive content over the Internet—Transmission Control Protocol (TCP). See EX1004, 2:42-62; 3:50-63. Duvall never mentions any other transport protocol, where network security threats also exist. EX2001, ¶191.

Network security systems often require inspecting raw packet data or at least a consideration of data from other alternative transport protocols—e.g., UDP. EX2001, ¶¶191-192; *see generally*, EX1014. Yet Duvall expresses no interest in monitoring anything else—and it would not make sense to use Duvall for that purpose either. EX2001, ¶¶191-192. Indeed, Duvall is unconcerned with the analysis of raw packet data. In fact, Duvall is only meant to operate on the presentation and application layers (layers 6/7), making Duvall wholly unsuitable for detecting network threats from any other layers. *Id.* While this would be

sufficient for censoring content, it is insufficient for network security monitoring.

Id.

Duvall also never contemplates a need for any sort of objective analysis to identify threats—which is the essence of network security. This is because Duvall assumes that the status of data as good or bad is subjectively predetermined by the user. EX1004, 2:12-16; *see also id.*, 3:25-29. So, instead of focusing on different analysis techniques or ways to detect unknown threats, Duvall focuses on different ways to censor the material already captured by its filters (*i.e.*, different blocking actions). *See, e.g., id.*, 3:20-41. Therefore, Duvall does not filter as part of any analysis related to security; it only filters to censor. *Id.*, 1:12-29; EX2001, ¶193.

Moreover, Duvall's disclosure is more concerned about an end-user bypassing the censorship than it is with any network security concerns (*i.e.*, it secures the network against internal users not external threats). *See, e.g.*, EX1004, 11:19-20. Indeed, Duvall is completely silent about any potential malware, viruses, or other network security-related concerns. Instead, Duvall devotes much of its disclosure to teaching how to implement patching code that cannot be easily disabled by the end-user. *Id.*, 10:35-11:19. But those concerns about the end-user are not the threats that a network security device is concerned of. From a network security point of view, Duvall's concerns are backwards—in a sense treating the end-user as the threat. EX1004, 10:34-11:19; EX2001, ¶195. Duvall's system is

therefore undesirable if the goal is network security. A POSA would not have been motivated to look for Chu nor combine it with Duvall to arrive at the security monitoring system claimed by Schneier.

2. Petitioner’s Motivation Relies on a Mischaracterization of Duvall’s Concerns about Subjectively “Filtering Too Much or Too Little”

Although Duvall discloses “filtering too much or too little,” (*i.e.*, not capturing all content that would be objectionable to a user or capturing non-objectionable content) this is only because the filtering criteria was not adequately configured. Duvall never suggests—as Petitioner implies—that its filtering systems would be inaccurate because the user was uncertain whether he believed something was objectionable. *See* Petition, 19. Nor does Duvall contemplate any techniques or analysis that could assist with that determination. EX2001, ¶197. Instead, Duvall’s concern is purely subjective—based on whether the “**user believes** that the filtering system is filtering too much or too little.” EX1004, 8:6-8 (emphasis added). Duvall just recognizes that the filters may need to be tuned to accommodate individual tastes and provides the solution accordingly. EX2001, ¶197. Duvall’s concern is whether its filters can accurately capture everything the user knows (*i.e.*, subjectively predetermined)—not identifying or resolving anything potential, uncertain, or unknown to the user.

Accordingly, Duvall provides an “editing manager” for users to do just that—tune the filters according to their subjective preferences. EX1004, 8:6-10. Duvall presents this as the solution to the problem of “filtering too much or too little.” *Id.* Given that Duvall provides the solution, a POSA would not actually seek any objective analysis techniques as Petitioner suggests. *See* Petition, 23-24, 38-41. Moreover, rather than introducing additional analysis techniques, Duvall’s solution embraces the subjective nature of its filtering system—simply allowing users to modify the filters accordingly. EX2001, ¶199. For Duvall, additional objective analysis makes no sense and would not resolve the problem actually contemplated by Duvall. *Id.* Indeed, Petitioner points to no objective techniques that could make subjective determinations on a user’s behalf. *Id.*

3. Duvall and Chu Use Filters in Different Ways and for Different Purposes

Petitioner relies on Chu’s whitelist and blacklist in its proposed combination to show that Chu is “like” Duvall and thus would be combined by a POSA. Petition, 21. But Duvall and Chu use their filters for different purposes. EX2001, ¶201. The sample Chu policy Petitioner relies on is meant to address issues of code-signing within trust management. EX1005, 43. Unlike Duvall, Chu’s REFEREE objectively determines the status of incoming data as trustworthy or untrustworthy using filters to determine whether an associated URL corresponds to

a known trusted/untrusted entity. *See id.*, 43-44; EX2001, ¶201. In contrast, Duvall merely filters to capture material that is “personally objectionable, whether that material is sexually explicit, violent, politically extreme, or otherwise, depending on that user’s individual tastes and sensitivities.” EX1004, 2:13-16.

So, whereas Chu uses filters to determine trust, Duvall only teaches a filtering system for censoring—but makes no objective determinations. Duvall never goes as far as using or suggesting how the filtering system could be used to perform any sort of analysis like identifying potentially security-related events. EX2001, ¶202. Instead, Duvall focuses on providing different ways to censor the content after a match with its filters (*i.e.*, different blocking actions).

4. A POSA Would Not Combine Chu with Duvall Given the Differences In Concerns, Underlying Assumptions, and Goals.

Duvall and Chu’s perspectives on “trust” are diametrically opposed. *See Id.*, ¶203. To the extent that Duvall has trust concerns, they are primarily with end-users bypassing the censorship. Duvall thus assumes the end-user is a “threat” and suggests several counter-measures such as password protecting the filter database and ensuring that even “if a[n] [end-]user kills the program by re-booting, the blocking mechanisms are left intact.” EX1004, 11:18-20. In contrast, Chu trusts the end-user—deferring ultimate decisions about whether to access a file back to the end-user when REFEREE cannot automatically decide. *See* EX1005, 43-44.

Because Chu is focused on trust management, its primary concern is whether to trust a third party. *See id.* Duvall lacks any shared concerns about trust and, likewise, Chu lacks the concerns found in Duvall. Given these differences, stemming from their differing goals, a POSA would not have been motivated to combine Chu with Duvall. EX2001, ¶¶203-204.

5. Even If a POSA Was Motivated to Search for Techniques to Resolve the Status of Residue, Chu Does Not Teach Any Such Techniques.

Petitioner asserts that a POSA would have been motivated to “seek techniques for more accurately resolving the status of residue data transmissions.” *Id.*, 20. But even if that were the case, Chu does not teach any such techniques. *See* EX1005, 43-44. Even applying Petitioner’s own rationale, a POSA still would not have been motivated to look to Chu to address this purported desire. EX2001, ¶206. Petitioner’s expert contends that Chu provides “techniques for determining whether to block or allow residue data.” EX1003, ¶104. But that statement is unsupported by Chu. At best, Chu provides techniques to determine the trustworthiness of incoming data using whitelists and blacklists. *See* EX1005, 43-44. But Chu is silent about any techniques to analyze residue from those filters. *See id.* Moreover, Petitioner argues the supposed desire for Chu comes from a desire to ensure Duvall’s filters are accurate—*i.e.*, to tune and edit the filters. *See* Petition, 19. But the policies Chu discloses are never used for that purpose. *See*

EX1005, 43-44. Nor does Chu apply “user intervention” for that purpose. *See id.*

Instead, Chu just teaches letting an end-user decide (*i.e.*, user intervention) whether to access a file. *See id.*

6. A POSA Would Not Have Arrived at the Claims Given Duvall and Chu.

Even with Duvall and Chu before him, there is simply no way for a POSA to arrive at the claimed invention. Petitioner’s rationale is internally inconsistent resulting in a proposed combination that makes little sense. Petitioner’s expert suggests a POSA would be motivated to “incorporate Chu’s user intervention into Duvall’s system for data transmissions not matching any of Duvall’s ALLOW or BLOCK (*i.e.*, positive or negative) filters, nor resolved by Duvall’s post-filtering residue analysis.” EX1003, ¶107. Petitioner modifies Duvall in at least three ways: to include an analyst and analyst system, transmit information to that analyst system, and then have that analyst provide feedback to the probe based on the analysis. But as explained, *supra* VII.B, Chu does not teach those features. Petitioner cannot just rely on hindsight and conclusory statements by its expert to argue that a POSA would find these modifications obvious. Petition, 21-23. Indeed, Chu never discloses the type of “user intervention” that Petitioner is attempting to introduce into Duvall and Petitioner’s resulting combination would be inoperable. EX2001, ¶¶210-220.

a. Chu Does Not Disclose the Type of “User Intervention”
that Petitioner Seeks to Introduce into Duvall

The “user intervention” Chu discloses is not as Petitioner represents. *Compare* Petition, 21-23, *with* EX1005, 44. Petitioner latches onto a single statement from Chu that “user intervention is needed only when the given URL is in neither the blacklist nor the whitelist” to suggest this somehow teaches transmitting residue to an analyst system for further analysis. *See* Petition, 21. But that is not what Chu means by “user intervention.” *See* EX1005, 44. According to Chu, when no filters are matched, REFEREE just returns control back to the application—prompting the end-user for attention. *See id.*, 32; 44. Chu merely recognizes the common-sense notion that filters can only make automatic decisions about data matched by the filters. EX2001, ¶ 210. Chu is not suggesting any further analysis or any intervention with respect to REFEREE.

b. Petitioner Applies “User Intervention” in Ways Never
Contemplated by Chu

Chu never applies “user intervention” the way Petitioner attempts modify Duvall. Petitioner applies “user intervention” on different data, for a different purpose, by a different human compared to what is taught by Chu.

First, Chu never uses “user intervention” on any identified potential events (*i.e.*, results of a first-level analysis). Instead, Chu’s “user intervention” only returns “*unknown*” after filtering the URL, before any first-level analysis could be

performed to identify potential events. EX2001, ¶212. Chu only makes a single trust/do-not-trust/unknown decision and thus is far from any second-level analysis using an analyst system.

Yet Petitioner attempts to incorporate Chu into Duvall to satisfy the limitations related to the second-level analysis. Petition, 19. But even under Petitioner's formulation of Duvall-Chu, this combination still makes no sense. Petitioner suggests Chu's "user-intervention" should be applied to data "not matching any of Duvall's positive/negative filters at step 104, nor resolved by Duvall's post-filtering residue analysis." Petition, 21. Petitioner also asserts that "information is only provided to the trained professional if the data is unresolved residue data—meaning that the data did not match any allow/block filters and passed through subsequent residue analysis (*e.g.*, keywords and pattern matching) without any resolution." Petition 40; *see also* EX1003, ¶146.

But even under Petitioner's formulation, the "unresolved residue data" cannot also be "an identified potential event." *See* Petition, 37-40. Though there is an amount of uncertainty tied to both types of data, no analysis has been performed on the purported "unresolved residue data" to elevate it from entirely unknown to the level of an identified potential event.

Second, Chu does not tune filters with "user intervention." Chu merely lets the end-user decide whether to access the file. *See* EX1005, 44. At best, this

relates to blocking or allowing but not tuning. If anything, Chu only properly replaces Duvall's default filters by letting the end-user decide what to see when no filters match with the content. But that would undermine Duvall's system (effectively letting the child control what should be censored).

Instead, Petitioner incorrectly argues Chu's "user intervention" is intended to help an administrator tune the filters more accurately. Petition, 19; 40. But Chu never teaches how this could be used to update filters—let alone to help an administrator tune them. EX2001, ¶216. On the contrary, the end-user's discretion to access a file is a very poor basis for tuning filters if security is truly a concern. *Id.*; *see also* EX1003, 108.

Schneier discloses remote analyst system at the SOC that assist in confirming whether the previously identified potential events are actually security-related, which makes sense because an analyst is better positioned to determine whether a potential event is actually a security threat compared to an end-user. EX1001, 2:18-24. Schneier provides the remote analyst system with access to more information gathered across various probes. *Id.*, 10:16-30. In contrast, Duvall's criteria are subjectively pre-defined, meaning trained security analysts (and analyst systems) are not even needed. EX1004, 2:12-16. Given Duvall's concerns about inefficiencies related to a remote database—remote administrators would make even less sense. *See id.*, 8:39-42. Moreover, because Petitioner

argues “information is only provided to the trained professional if the data is unresolved residue data,” Petition 41, such administrators would not even have the information necessary to tune pre-existing filters—only add new ones, EX2001, ¶218.

Third, Petitioner asserts that “user intervention” should be combined into Duvall such that it is performed by an analyst rather than the end-user. Petition, 21-23; 39-40. But this is inconsistent with and not taught by Chu. Petitioner recognizes Chu does not mention any analyst or analyst system, and thus argues Duvall’s filtering system would be controlled by a remote administrator. *See* Petition 22. But Petitioner never explains how “user intervention”—as taught by Chu—could assist administrators. Instead, Chu’s intervention is by the end-user. *See* EX1004, 44. If “user intervention” involved an administrator instead, Chu’s invention would fall apart due to the flood of prompts faced by the administrator. EX2001, ¶220. Therefore, requiring intervention by an administrator is inconsistent with and never suggested by Chu.

c. Petitioner’s Resulting Combination Would Be Inoperable

Petitioner overlooks the importance of limiting what is sent to the analyst system in Schneier—resulting in an inoperable combination. *Id.*, ¶221. Indeed, Schneier recognizes that the volume of residue data is unmanageable without first

identifying the data that rises to the level of a potential event (to narrow what should be sent to the analyst system). *See, e.g.*, EX1001, 2:5-12, 1:36-39.

Yet Petitioner’s resulting combination sends an administrator information about every transmission “not matching any of Duvall’s positive/negative filters at step 104, nor resolved by Duvall’s post-filtering residue analysis.” Petition, 21. But because Duvall never narrows down what should be sent like Schneier, Duvall’s system just results in the same flood of information that Schneier is designed to prevent. EX2001, ¶222. Indeed, regardless of whether Duvall-Chu can actually be combined or characterized as Petitioner suggests, the resulting combination Petitioner proposes would still flood the analyst system with an unmanageable amount of information. *Id.*, ¶223. Therefore, as a practical matter, such a combination would be inoperable. *Id.* It faces the exact problem that Schneier’s secondary considerations addressed. *Id.*

There is no way to combine these references to arrive at these aspects of the claimed invention. *Hamilton Beach Brands, Inc. v. f’real Foods, LLC*, 908 F.3d 1328, 1341-42 (Fed. Cir. 2018).

D. There Is No Way to Combine Cogger into Duvall-Chu To Arrive At Independent Claim 18

There is no motivation to combine Cogger into Duvall-Chu. Nor is there a logical reason to use a remote interface for a pre-existing telecommunications

trouble ticketing system for Duvall-Chu. Cogger simply allows customers to remotely access a pre-existing system through its disclosed interface. *See id.*; EX2001, ¶226. But accessing a system remotely is not the same as using the underlying system to remotely manage a filtering database. Yet Petitioner modifies Duvall-Chu to do just that. Petition, 65-67.

Petitioner never explains why a POSA would look at Cogger in the first place. Cogger is concerned with a legacy telecommunications trouble ticketing system that Duvall-Chu does not already have. *See* EX2001, ¶227. And Petitioner never explains how Cogger would be adapted for use with Duvall-Chu, much less in a network security context. Nor does Cogger disclose the specifics of the ticketing system so that a POSA could implement one for Duvall-Chu. *See Id.*, ¶227. Instead, Cogger only teaches the interface to access that telecommunications ticketing system's pre-existing features. *See* EX1033, 2:39-43.

Moreover, Petitioner's combination first requires Duvall-Chu to utilize remote management before a trouble ticketing system for that remote management even begins to make sense. But that teaching is absent from Duvall, Chu, and Cogger. Instead, Petitioner cites to (but does not seem to rely on) yet another reference. *See also* Petition, 22, 45. Petitioner never asserts that other reference forms part of its unpatentability grounds, provides no insight into how it could be done, and offers no motivation to do so. *See* EX2001, ¶228.

Petitioner also provides no details about how Cogger could even be combined into Duvall-Chu with a reasonable expectation of success. Cogger relies on a complex pre-existing architecture and a suite of other pre-existing telecommunications applications. *See* EX1033, 1:32-36; 6:8-14; 13:66-14:17; *see also* EX2001, ¶229. None of these systems exist in Duvall-Chu. A POSA could not just adapt the pre-existing telecommunications trouble ticketing system—nor the entire suite of applications Cogger requires—into Duvall-Chu with any reasonable expectation of success. *Id.*, ¶230. Merely implementing the user interface taught by Cogger would accomplish nothing. *Id.*

VIII. CONCLUSION

Petitioner's references fail to disclose multiple claim limitations. And Petitioner fails to show how its references could be combined into an operable whole, much less without the use of impermissible hindsight and with a reasonable expectation of success. The PTAB should deny institution.

IPR2023-00889
U.S. Patent No. 7,895,641

Respectfully submitted,

Dated: August 29, 2023

/Nolan M. Goldberg/

Nolan M. Goldberg (Reg. No. 54,126)
Baldassare Vinti (Reg. No. 43,970)
PROSKAUER ROSE LLP
Eleven Time Square
New York, NY 10036-8299

Jonathan A. Roberts (Reg. No. 68,565)
Raymond Y. Mah (Reg. No. 41,426)
Joseph A. Rhoa (Reg. No. 37,515)
NIXON & VANDERHYE P.C.
901 N. Glebe Road, 11th Floor
Arlington, VA 22203-1808

*Counsel for Patent Owner
BT Americas Inc.*

IX. WORD COUNT CERTIFICATION

Under the provisions of 37 C.F.R. § 42.24(d), the undersigned hereby certifies that the word count for the foregoing Patent Owner's Preliminary Response totals 13,999 words, which is less than the 14,000 allowed under 37 C.F.R. § 42.24. Counsel has relied upon the word count feature provided by Microsoft Word.

Dated: August 29, 2023

/Nolan M. Goldberg/

Nolan M. Goldberg (Reg. No. 54,126)
PROSKAUER ROSE LLP
Eleven Time Square
New York, NY 10036-8299

*Counsel for Patent Owner
BT Americas Inc.*

X. CERTIFICATE OF SERVICE

The undersigned hereby certifies that a true copy of the foregoing Patent Owner's Preliminary Response and all exhibits filed therewith, was served on Petitioner on August 29, 2023, by emailing a copy to counsel at the email addresses listed below:

Jonathan Tuminaro
Daniel S. Block
Michael D. Specht
Steven M. Pappas
STERNE, KESSLER, GOLDSTEIN & FOX P.L.L.C.
1100 New York Avenue, N.W.
Washington, D.C. 20005
Tel. (202) 371-2600
Fax. (202) 371-2540

jtuminar-PTAB@sternekessler.com
mspecht-PTAB@sternekessler.com
dblock-PTAB@sternekessler.com
spappasPTAB@sternekessler.com
PTAB@sternekessler.com.

Dated: August 29, 2023

/Jonathan A. Roberts/

Jonathan A. Roberts
NIXON & VANDERHYE P.C.
901 N. Glebe Road, 11th Floor
Arlington, VA 22203-1808
Registration No. 68,565

*Counsel for Patent Owner
BT Americas Inc.*