IN THE UNITED STATES DISTRICT COURT
FOR THE EASTERN DISTRICT OF TEXAS
MARSHALL DIVISION

SECTRA COMMUNICATIONS AB, Plaintiff, v. CISCO SYSTEMS, INC. Defendant.	Civil Action No. 2:26-cv-00174-RWS (LEAD CASE) Jury Trial Demanded
SECTRA COMMUNICATIONS AB, Plaintiff, v. FORTINET, INC. Defendant.	Civil Action No. 2:26-cv-00175-RWS Jury Trial Demanded

PLAINTIFF SECTRA COMMUNICATIONS AB'S MOTION TO COMPEL COMPLIANCE WITH PROTECTIVE ORDER

TABLE OF CONTENTS

I. DEFENDANTS ARE UNREASONABLY WITHHOLDING APPROVAL. 1

A. The Objections That Defendants Seemingly Abandoned. 1

B. The "Breaking Out of the Sandbox" Objection Is Technically Unfounded. 3

C. The "Tools Can Change Read-only Code" Objection Is Technically Unfounded. 5

II. DENYING THE TOOLS WOULD PREJUDICE SECTRA AND CONFLICT WITH THE PROPORTIONALITY PRINCIPLES OF RULE 26. 6

III. DEFENDANTS' OWN AI PROWESS UNDERMINES THEIR OBJECTIONS. 7

IV. CONCLUSION 7

TABLE OF AUTHORITES

Rules, Statutes, and Other Authorities

Fed. R. Civ. P. 26(b)(1) 6

Plaintiff Sectra Communications AB respectfully moves the Court to compel Defendants Cisco Systems, Inc. and Fortinet, Inc. to comply with the Protective Order, which permits Sectra to request source code review tools and provides that Defendants cannot unreasonably withhold approval. ECF 38, ¶ 10(e)(ii). Defendants are violating this Order by unreasonably withholding approval for several such tools. The Court should compel Defendants to install the tools requested so that code review may proceed expeditiously.

I. DEFENDANTS ARE UNREASONABLY WITHHOLDING APPROVAL.

The PO provides that Sectra may request "licensed software … to assist with [source code] review" and that Defendants cannot unreasonably withhold approval. ECF 38, ¶ 10(e)(ii). Sectra requested, among other tools, Cline, Ollama, Qwen, Python, and Node.js. Some of these tools (Cline, Ollama, and Qwen) include or leverage large language models (sometimes called "AI") solely within the review computer. Ex. 1, ¶¶ 2-4. There is no dispute that each of these tools that Sectra requested "assist with review." *Id*., ¶¶ 2-5, 12. Defendants have nevertheless refused their consent without offering a reasonable basis for doing so. Indeed, Defendants' objections have shifted throughout the meet-and-confer process; each time Sectra addresses an objection, Defendants abandon it and raise a new one. *See* Ex. 2, ¶¶ 2-3. At this time, Sectra understands that Defendants are standing on two objections. But even this is unclear. When pushed to provide a written explanation to substantiate their final reasoning for withholding approval, Defendants refused. *Id*.

A. The Objections That Defendants Seemingly Abandoned

Defendants initially claimed that the PO limits Sectra to tools for "viewing and searching source code," and they objected to the subject tools because they are capable of doing more than that. *Id*. But this interpretation of the PO is wrong, as Sectra explained to Defendants. Paragraph 10(e)(ii) explicitly permits Sectra to request tools "to assist with review," which it has done. Paragraph 10(e)(iii) separately addresses the narrower scenario in which the requested tools are

"commercially available," requiring Sectra to provide a licensed copy. ECF 38. There is no "viewing and searching" limitation expressed in those provisions; if there were, even the most basic review tools would not comply, as they all are capable of doing more than "viewing and searching" code. *See* Ex. 1, ¶¶ 7-8, 10. Nevertheless, as Sectra also explained, the purpose of the requested tools is to assist Sectra with reviewing and searching the code; not compiling, executing, or modifying it. *Id.*, ¶¶ 2-5. Sectra even expressly agreed in writing that it would not compile any code. Ex. 2, ¶ 2. Defendants have not raised this objection since.

Defendants then claimed that the requested tools may ingest and process the code, create embeddings or other derived data, generate code or summaries, and retain derived content or information outside the original files. Ex. 2, ¶ 2. But as Sectra explained—and Defendants never rebutted—these concerns describe the basic operation of virtually every search and review tool, including those not objected to. *Id.*; Ex. 1, ¶ 7. The *only* difference between the tools Defendants approved and the AI tools at issue "is the analytical method"; and there is *no* difference in this regard between the approved tools and Python and Node.js, which have been around for many years. Ex. 1, ¶ 7. Accordingly, as Sectra pointed out, Defendants' position would logically require them to object to every tool on the list, not just the ones they have singled out. Ex. 2, ¶ 2. Defendants have not objected to every tool, revealing that the objection is arbitrary and unreasonable. And, notably, Defendants have not raised this particular objection again since Sectra pointed this out.

To date, as far as Sectra understands, these two objections discussed above are the only objections that Defendants have levied against Python and Node.js. Defendants have not substantiated these objections at all since Sectra provided detailed explanations rebutting them. It thus remains unclear why Defendants continue to object to Python and Node.js.

Finally, Defendants later objected to Qwen in particular because it was developed in China. *Id*. So, at Defendants' request, Sectra identified three acceptable alternative models developed in

other countries, namely, North Mini Code, Devstral Small, and Granite. *Id*. Defendants nevertheless continue to withhold approval for the requested suite of tools. *Id*.

B. The "Breaking Out of the Sandbox" Objection Is Technically Unfounded.[1]

One of Defendants' current objections—that the requested AI tools could "break out of the sandbox" (i.e., access a network without permission)—ignores reality and reflects a fundamental misunderstanding of how these tools operate. As a threshold matter, if the source code computer is not physically capable of connecting to a network, it is impossible for the software on the computer to go anywhere else. Ex. 1, ¶ 9. For computers without Wi-Fi, this is accomplished by not connecting an Ethernet cable to the computer. *Id*. For Wi-Fi-enabled computers, Wi-Fi can be physically disabled by, depending on the machine, flipping a switch, removing the Wi-Fi card, and/or removing Wi-Fi antennas. *Id*. Moreover, Wi-Fi capabilities can be totally and reliably disabled via tried-and-true software avenues, including BIOS/UEFI configurations and disabling the wireless adapter in the computer's Device Manager. *Id*. It is not an overstatement to say that one or more of these methods are used in *every single case involving source code analysis* to prevent a source code computer from connecting to a network. *Id*., ¶ 13. What's more, Defendants are industry leaders in cybersecurity, so color Sectra skeptical of Defendants' objection considering they are uniquely capable of ensuring that a review computer cannot impermissibly connect to a network. Put simply, Defendants need only provide a review computer that cannot connect to the Internet to assuage this concern. *Id*.

Sectra suspects that Cisco will argue that these safeguards are not sufficient because it has elected to produce its source code via a network connection, specifically, via a remote thin client connected to a virtual machine maintained behind Cisco's firewall,[2] which is one of the two methods

[1] Sectra does not believe that Defendants contend that this objection applies to Python and Node.js because they are not "AI tools."

[2] This essentially means that Cisco will provide a source code review computer that, instead of having Cisco's source code stored on it, will have limited network connectivity to access a remote,

authorized by Paragraph 10(a) of the Protective Order. This choice is fatal to Cisco's "breaking out of the sandbox" objection for two reasons. *First*, Cisco *chose* this production method. It could just as easily produce the code in an unnetworked, local manner, as Fortinet is doing, which would completely mitigate any "breaking out of the sandbox" risk as discussed above. Cisco cannot use its own choice of production methodology to deprive Sectra of legitimate review tools. *Second*, Cisco is in a far *better* position to monitor and control what happens via the proposed setup than it would be on a standalone machine. Indeed, Cisco can impose whatever restrictions it deems appropriate at the hypervisor, operating-system, or firewall level to contain the tools' operations. *Id*., ¶ 9. The thin-client architecture gives Cisco more control, not less—which is precisely why Cisco chose it, and precisely why Cisco's purported concerns are baseless. *Id*.

Moreover, the technical reality of how the requested tools function reveals that Defendants' objections are unfounded. Defendants essentially allege that the tools (namely, Sectra believes, Ollama) might reach outside the review environment. But Ollama and the other tools only run locally, with Internet access blocked and no source code or other data being transmitted. *Id*., ¶¶ 2, 4, 6. On a standalone machine with no network interface (Fortinet's setup), there is literally nowhere for data to go. *Id*., ¶ 7. On Cisco's thin-client setup, Cisco controls the firewall and can trivially block any inbound/outbound traffic to/from any processes. *Id*. And the requested models are openly published, meaning that Defendants can examine the model architecture and verify exactly what is being installed. *Id*., ¶ 11. This level of auditability and transparency exceeds what is available for most proprietary commercial software tools. *Id*.

In short, the "sandbox" concern fails. Under Cisco's architecture it fails because Cisco completely controls the network environment, and Cisco could simply provide a regular review

---

siloed environment (virtual machine) set up by Cisco that holds the source code.

computer instead. Under Fortinet's architecture it fails because there is no network at all.

C. The "Tools Can Change Read-only Code" Objection Is Technically Unfounded.[3]

Defendants' other objection is that even if the code is produced as read-only (which Sectra agreed to), the requested AI tools could nevertheless "change the code." This fails for three reasons.

First, it is technically incorrect. Read-only permissions are enforced at the operating-system level and make modifying files impossible, and no application—AI-powered or otherwise—can override those protections. Ex. 1, ¶ 10. Indeed, approved review tools have editing and execution capabilities that are neutralized by such protections; the requested tools have comparable capabilities that are nullified in the same manner. *Id*. There is no mechanism by which the requested tools can bypass OS-level access controls. *Id*. Put simply, if the source code is provided as read-only, it remains read-only regardless of what software is running on the review machine.

Second, the tools that Defendants already approved pose equal or greater security risks than the tools they have rejected. *Id*., ¶¶ 7, 10. If Defendants' security concerns were genuine, they would have objected to these commonly-used tools as well. That they approved these tools while objecting to a local language model that processes queries and returns results reveals that the objection is not grounded in any actual security or confidentiality risk. *Id*., ¶¶ 3, 7-8, 13.

Third, and finally, even assuming the read-only source code was somehow modified (by the approved or unapproved tools), there is a simple way to mitigate this risk: compare the original code with the reviewed code. *Id*., ¶ 10. Indeed, there are simple, straightforward ways to do this, including Git (a software tool used by nearly every developer) and checksums (an algorithm that takes a mathematical calculation of the original text and compares it to the calculation of the potentially-modified text to see if it matches). *Id*. And, again, Defendants are industry leaders in cybersecurity;

[3] Again, because Python and Node.js are not "AI tools," Sectra does not believe that Defendants contend that this objection applies to those tools.

it defies logic to think that they are not uniquely equipped to determine whether their own code has been modified. In short, the risk of modifying read-only code is non-existent. To the extent there is a risk, it certainly is not unique to the subject tools. If Defendants are legitimately concerned about this, there is a surefire, straightforward way address such concerns. This is not a reasonable objection.

II. DENYING THE TOOLS WOULD PREJUDICE SECTRA AND CONFLICT WITH THE PROPORTIONALITY PRINCIPLES OF RULE 26.

Contentions must be supplemented with source code within 30 days after it is produced. ECF 37. The repositories are likely to be enormous, and the requested tools offer very significant efficiencies—they are, quite simply, the latest and greatest review tools available. Ex. 1, ¶ 12. Without them, the review will be considerably more expensive and time-consuming, and it will place substantial additional pressure on Sectra's source code review. *Id*. This, of course, appears to be Defendants' true motive, and forcing Sectra to conduct a review of what are likely massive source code repositories without these tools is disproportionately burdensome and inconsistent with the principles of Federal Rule of Civil Procedure 26(b)(1). This is particularly true when there is a simple, easy way to address all of Defendants' stated concerns: **put "read-only" source code on a computer with no Internet connectivity**. *Id*., ¶ 13. This would address all of Defendants' issues and amount to no extra burden at all; but the benefit to Sectra would be significant. *Id*.

It bears mention that Dr. Shoemake is currently using these exact AI tools (with the Chinese-based Qwen model, no less) to analyze source code in another EDTX case, and no objection was raised to their installation or use. *Id*., ¶ 12. At bottom, the tools exist, Sectra has requested them, they pose no additional risk beyond the tools Defendants have already approved, and there is a simple way to address Defendants' concerns. *Id*. Withholding approval of the requested tools serves no purpose other than to make Sectra's review slower, more expensive, and less thorough.

III. DEFENDANTS' OWN AI PROWESS UNDERMINES THEIR OBJECTIONS.

For all of Defendants' bluster about the risks that AI tools pose to their source code, they actively develop and deploy their own AI products and services.[4] Their willingness to leverage AI when it benefits them and their customers, while prohibiting a patent plaintiff from using AI to review their source code, underscores the unreasonableness of their position. Both Cisco and Fortinet also promote their prowess in controlling and securing AI activities for businesses.[5] If there is anyone capable of ensuring that an AI-adjacent review tool—particularly one siloed in a non-networked environment and only given read-only code—is configured safely, it is Defendants.

IV. CONCLUSION

For the foregoing reasons, Sectra respectfully requests that the Court compel Defendants to install the requested review tools (Ollama, Cline, North Mini Code 1.0, Devstral Small 2, Granite 4.2 30B, Python, and Node.js) on the Source Code Computer(s), subject to the safeguards already provided by the Protective Order and the additional safeguards Sectra has proposed and agreed to.

[4] https://www.cisco.com/site/us/en/products/security/secure-access/ai-access/index.html; https://www.cisco.com/site/us/en/solutions/artificial-intelligence/ai-assistant/index.html; https://www.fortinet.com/solutions/ai-security.

[5] *Supra* n.2.

Dated: September 16, 2026

Respectfully submitted,

/s/ Carder W. Brooks

Edward R. Nelson III
State Bar No. 00797142
ed@nelbum.com
Brent N. Bumgardner
State Bar No. 00795272
brent@nelbum.com
Christopher G. Granaghan
State Bar No. 24078585
chris@nelbum.com
Carder W. Brooks
State Bar No. 24105536
carder@nelbum.com
Jinming Zhang
Virginia Bar No. 96210
jinming@nelbum.com
NELSON BUMGARDNER CONROY PC
3131 West 7th Street, Suite 300
Fort Worth, Texas 76107
817.377.9111

COUNSEL FOR PLAINTIFF
SECTRA COMMUNICATIONS AB

CERTIFICATE OF SERVICE

I hereby certify that a copy of the foregoing document was filed electronically in compliance with Local Rule CV-5(a). Therefore, this document was served on all counsel on September 16, 2026.

/s/ Carder W. Brooks

CERTIFICATE OF CONFERENCE

I hereby certify that counsel for Plaintiff and counsel for Defendants have complied with the meet and confer requirements in Local Rule CV-7(h), and the relief sought by this motion is opposed. The conferences required by the rule were conducted on August 31, September 11, and September 15. Edward R. Nelson III and Carder W. Brooks attended the September 15 conference for Plaintiff, and Carder W. Brooks attended the August 31 and September 11 conferences for Plaintiff. Matt Gaudet attended the September 15 conference for Defendants. My understanding is that David Dotson and Alice Snedeker attended the August 31 and September 11 conferences for Defendants. No agreements could be reached on the conferences for the reasons set forth in this motion. Discussions have conclusively ended in an impasse, leaving an open issue for the court to resolve.

/s/ Edward R. Nelson III