ה-SOC שלא ישן: סוכן קולי שחוקר ומתקן בזמן אמת

TLV-DEM307 — סיכום סשן, AWS Summit Tel Aviv 2026

יניב קדוש, Solutions Architect, ושימי רוקח, Senior Solutions Architect, AWS

10 בספטמבר 2026 | משך: כ-24 דקות | הופק מהקלטת הסשן

סיכום מאת קובי אלמוג

על המסמך הזה

המסמך מסכם את הסשן TLV-DEM307, "Always On Call: Real-Time Security Response with AWS and Voice AI", מתוך מסלול ה-AWS Demos ב-AWS Summit Tel Aviv 2026. הוא נבנה מתמלול אוטומטי של ההקלטה המלאה ומהסליידים ומסכי הדמו שחולצו מהווידאו עצמו, כך שאפשר לחזור לתוכן — ובעיקר לדמו החי — בלי לצפות שוב בעשרים וארבע הדקות. חותמות הזמן בגוף המסמך מפנות לנקודה בהקלטה.

איך לקרוא את זה

- **פרק 1 — תקציר מנהלים:** ארבע דקות קריאה. מה הוצג בפועל ומה שווה לבחון.
- **פרקים 2–4 — המסגרת:** המספרים שמצדיקים את הפתרון, שלוש ההבטחות, וארבעת השירותים שמהם הכול בנוי.
- **פרקים 5–7 — הארכיטקטורה והדמו:** מה בדיוק רץ על הבמה, ומה ה-agent הקולי אמר ועשה.
- **פרקים 8–10 — מאחורי הקלעים:** Security Hub Exposure, Durable Lambda עם callback, והקוד של הסוכן.
- **פרק 11 — מה לוקחים מכאן:** מסקנות, מגבלות ונקודות להמשך.

הערת מקור המסמך הופק מתמלול אוטומטי של ההקלטה. התמלול סביר בקטעים המקצועיים, אבל מונחים לועזיים הופיעו בו בשיבוש פונטי ("אג'נט קור" = AgentCore, "סטרנד"/"Strands" = Strengths, "דיורה בלמדה" = Durable Lambda) והם תוקנו כאן מול הסליידים. הציטוטים אומתו מול ההקלטה בחותמת הזמן המצוינת ומובאים כלשונם, כולל שיבושי ASR קלים; קטעים שנאמרו באנגלית נשארו באנגלית. כל המספרים והשמות במסמך מגיעים מהסליידים או מדברי הדוברים.

סליד הפתיחה: קוד הסשן DEM307 ושמות שני הדוברים, שניהם Solutions Architects ב-AWS.

1. תקציר מנהלים

יניב קדוש ושימי רוקח, שני Solutions Architects מצוות הסטארטאפים של AWS שמתמחים בסייבר, לקחו שאלה אחת — למה חצי מהאלרטים בצוותי SOC לא נוגעים בהם — ובנו עליה דמו חי אחד: מהנדס אבטחה מקבל התראה בסלאק, פותח דפדפן, ומדבר בקול עם סוכן AI שמסביר לו מה נחשף, מה ההשלכה, ומציע תיקון — ומבצע אותו רק אחרי אישור קולי מפורש. הסשן היה 90% דמו ו-10% מצגת, והדמו רץ בשידור חי על חשבון AWS אמיתי.

- **הבעיה שהוצגה היא קיבולת, לא זיהוי.** שני מספרים מהסליד הפותח: מעל 50% מהאלרטים לא מטופלים כלל, ו-257 יום בממוצע לזהות ולהכיל אירוע אבטחה (מקורות: IBM ו-Morning Consult). שימי: "אובר עול 50% לא מספיקים להגיע אליהם בכלל שאין מספיק כוח אדם" [0:00].
- **הממשק הוא קול, לא דשבורד.** הטענה המרכזית: את ה-triage עושים בשיחה בשפה טבעית, לא בקפיצה בין טאבים של לוגים — "אנחנו הולכים לעשות את האינבסיגציה באנגלית לא עם דאצ'בורדים" [3:02]. הבחירה ב-Amazon Nova 2 Sonic היא בחירת latency: מודל speech-to-speech ולא cascade של speech-to-text.
- **human-in-the-loop הוא לא סיסמה אלא רכיב ארכיטקטוני.** העצירה לאישור אנושי ממומשת ב-AWS Lambda Durable Functions דרך קריאת wait_for_callback: הפונקציה נעצרת, לא צורכת compute ולא מחויבת, וממתינה ל-callback ID שהסוכן יקרא לו רק אחרי אישור מילולי.
- **הממצא שטופל הוא Exposure, לא finding בודד.** Security Hub מתאם בין API Gateway ללא אותנטיקציה לבין Lambda עם רול מוגזם, והופך את הצירוף ל-exposure יחיד עם attack path ו-blast radius — הגרף שהוצג הוליך מ-soc-demo-exposed-lambda דרך שלוש פוליסות FullAccess אל דליי S3, טבלאות DynamoDB, EC2 ו-ECR.
- **הכול AWS-native וזמין היום.** ארבעה שירותים בלבד: Amazon Nova 2 Sonic, Amazon Bedrock AgentCore Runtime, AWS Lambda Durable Functions (שהוכרז ב-re:Invent 2025) ו-Amazon Security Hub. אין רכיב צד שלישי בארכיטקטורה.
- **המגבלה שהודו בה מעל הבמה: עברית עדיין לא נתמכת.** הדמו כולו התנהל באנגלית; תמיכה בעברית הוזכרה כ-roadmap, והשפות הקיימות שנמנו הן אנגלית, צרפתית וספרדית עם מגוון מבטאים.

2. הבעיה: צוותי SOC ללא קיבולת

שימי רוקח פתח בשני מספרים בלבד, ובמפורש הציג אותם כ"המחשה" ולא כמחקר. הראשון: כ-50% מהאלרטים שמגיעים לצוותי SOC לא מטופלים. יש תעדוף, והדברים הקריטיים כן נוגעים בהם, אבל בשורה התחתונה "אובר עול 50% לא מספיקים להגיע אליהם בכלל שאין מספיק כוח אדם" [0:00].

שני המספרים שפתחו את הסשן, עם המקורות מצוינים בתחתית הסליד.

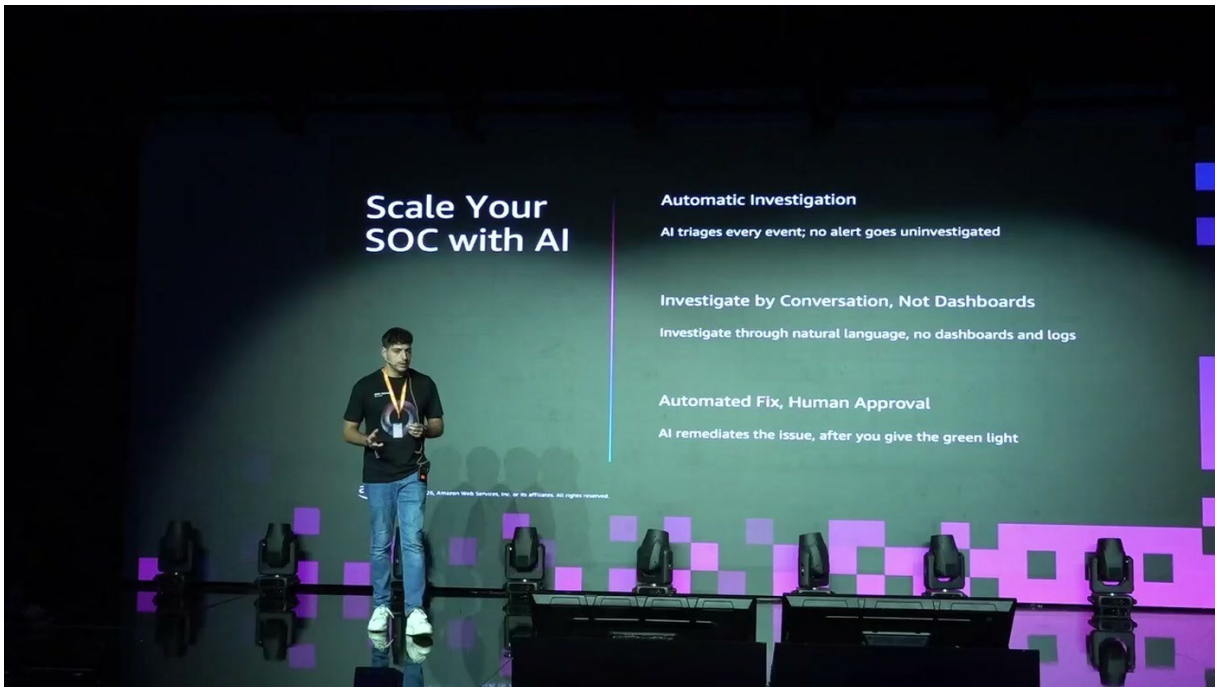
המספר השני נוגע לזמן ולא לכמות: "כשיש איזשהו בריץ בארגון לוקח בממוצע כ257 יום לסגור אותו מקצה לקצה" [1:30] — כולל החקירה, ההבנה מה דלף ואיפה, והסגירה. גם כאן הוא סייג שבאירוע קריטי הזמן קצר בהרבה, ושמדובר בממוצע.

המסקנה שהוא גזר משניהם הייתה חד-משמעית: "צוותי הסוק שלנו עמוסים" [1:30]. הסיבה, לטענתו, אינה חוסר בכלי גילוי אלא עודף שלהם, בשילוב עם צד ההתקפה: "היום בעידן האיי קל הרבה יותר למצוא פסים להתקפה, ולכן יש הרבה יותר מתקפות והרבה יותר אירועים" [1:30]. ומכאן הניסוח של האתגר — למצוא דרך לעשות scale לצוותים האלה "בלי רק לסמוך על הוספת עוד אנשים לצוות" [1:30].

> **זווית, לא מוצר** שימי הקדים והבהיר שהסשן אינו סקירת שוק: "כמובן יש מוצרים בשוק שעוזרים לצוותי סוק, אבל אנחנו לוקחים איזושהי זווית מאוד שונה וייחודית" [1:30]. כלומר, מה שהוצג הוא דמו רפרנס לבנייה עצמית, לא המלצה להחליף פלטפורמת SOAR קיימת.

3. שלוש ההבטחות

יניב קדוש הציג את המסגרת בשלושה סעיפים, שהם בפועל שלד הדמו כולו.



שלושת עקרונות הפתרון: חקירה אוטומטית של כל אירוע, חקירה בשיחה במקום בדשבורדים, ותיקון אוטומטי בכפוף לאישור אנושי.

- **Automatic Investigation.** ה-AI עושה triage לכל אירוע, כך שאף אלרט לא נשאר לא-נחקר. הטענה המעשית: לא עוד "אני אטפל בזה ביום ראשון" [1:30], אלא טיפול באותו יום.
- **Investigate by Conversation, Not Dashboards.** "אנחנו הולכים לעשות את האינבסיגציה באנגלית לא עם דאצ'בורדים לא בטריאג' במלא לוגים טאבים" [3:02]. הסוכן מתפקד, במילותיו, כמו מפתח סניור שמסביר — "אפילו אם זה שתיים בבוקר" [3:02].
- **Automated Fix, Human Approval.** כאן הוא נעצר במפורש: חשוב להשאיר human in the loop בארכיטקטורה אג'נטית, "שאנחנו יכולים לאשר לאי איי להתקן שלא ישבור כלום בתוך פרודקשן או לעשות משהו שאנחנו לא רוצים שזה יעשה" [3:02].

4. ארבעת השירותים

הפתרון כולו נשען על ארבעה שירותים, וכל אחד מהם נבחר מסיבה שהוצהרה על הבמה.

ארבע אבני הבניין, מסווגות לפי תפקיד: Voice AI, בינה מלאכותית, פונקציה, ואבטחה.

Amazon Nova 2 Sonic

מודל ה-foundation של AWS ל-speech-to-speech. ההבחנה שיניב הדגיש היא ארכיטקטונית ולא שיווקית: "זה speech-to-speech זה לא ה-old cascaded model שזה speech-to-text" [3:02], והנימוק הוא latency — בשרשור של תמלול, מודל טקסט וסינתזה נצבר עיכוב שהורג שיחה טבעית.

Amazon Bedrock AgentCore Runtime

הסוכן רץ בקונטיינר על AgentCore Runtime. את השאלה "למה לא EKS או ECS" הוא ענה מראש: "כי אנחנו יכולים לעשות את ה-deployed production בצורה ממש יותר מהר" [4:33]. מה שה-runtime מביא מוכן: gateway ו-authorization לחיבור ל-MCPs, APIs וטולים אחרים, observability, security ו-identity — ובנוסף infrastructure ו-capacity שמנוהלים על ידי AWS, "אנחנו לא צריכים לדאוג על capacity אנחנו דואגים על ה-innovation" [4:33].

AWS Lambda Durable Functions

הרכיב שמנהל את ה-workflow. שלושת ההבדלים מ-Lambda רגילה שנמנו: זה multi-step workflow, זה stateful, וזה "יכול לרוץ יותר מ-15 דקות הוצאנו את זה ב-reinvent 2025" [4:33]. כאן יושב ה-remediation playbook.

Amazon Security Hub

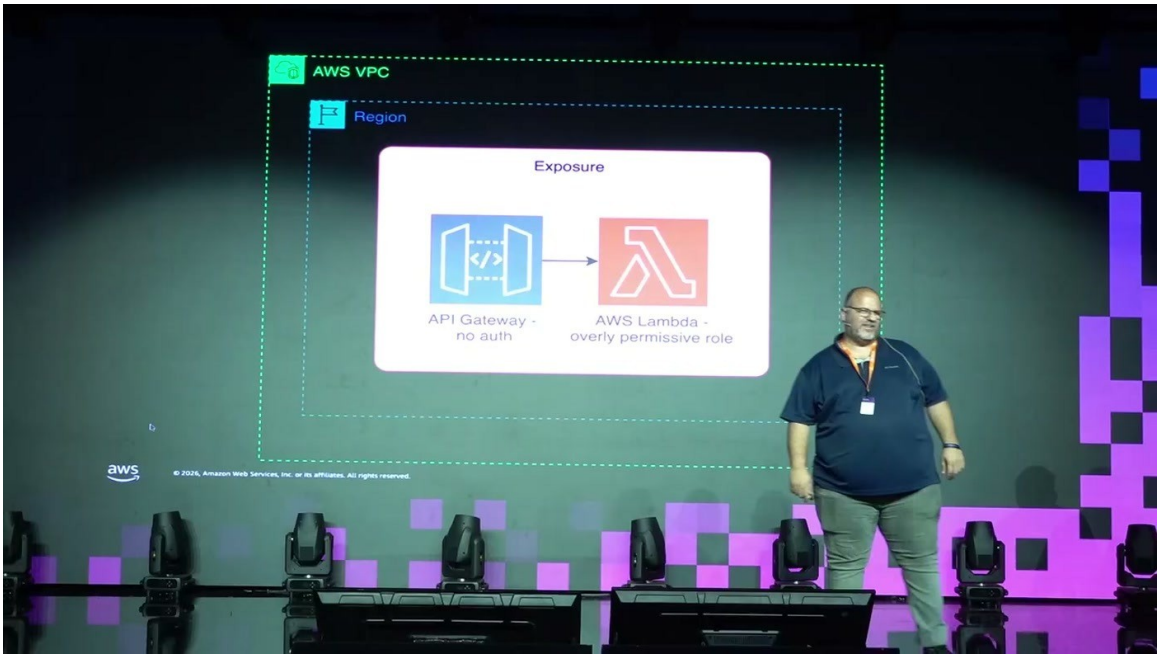
מקור הממצאים. "זה ה-single pane of glass שאתם יכולים למצוא את ה-findings וה-exposure" [6:05], וה-exposure הוא שמראה את ה-full attack path בצורה ויזואלית.

5. הארכיטקטורה

שימי עבר על הארכיטקטורה מהר במכוון — "באנו לראות דמו ולא מצגת" [6:05] — בשלושה סליידים מצטברים.

החשיפה שנבנתה במכוון

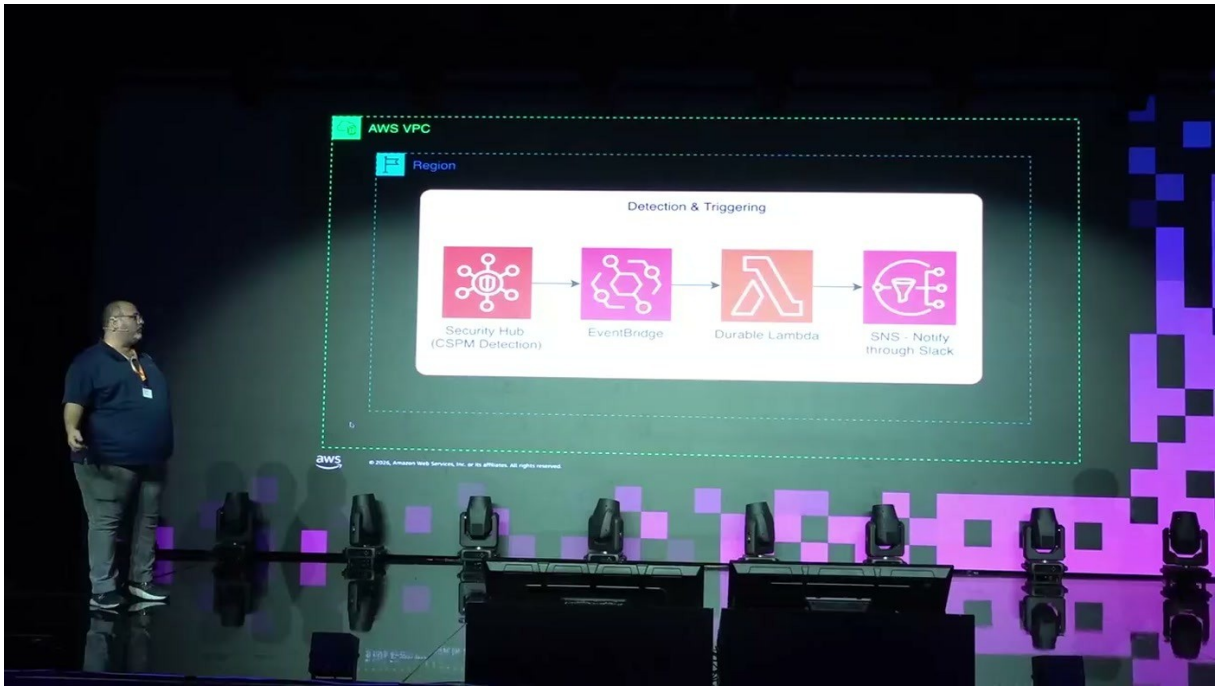
הצעד הראשון בסשן היה ליצור exposure אמיתי בחשבון: "יש לנו API Gateway, אין אותנטיקציה למתודה שם והיא מטריגה פונקציית למדה שהיא overly permissive, יש עליה admin access לכל מיני שירותים ב-AWS" [6:05].
השילוב של השניים — ולא כל אחד לחוד — הוא מה ש-Security Hub מזהה.



שני הרכיבים שיוצרים יחד את החשיפה: API Gateway ללא אימות, ומאחוריו Lambda עם רול מרשה מדי.

זיהוי והפעלה

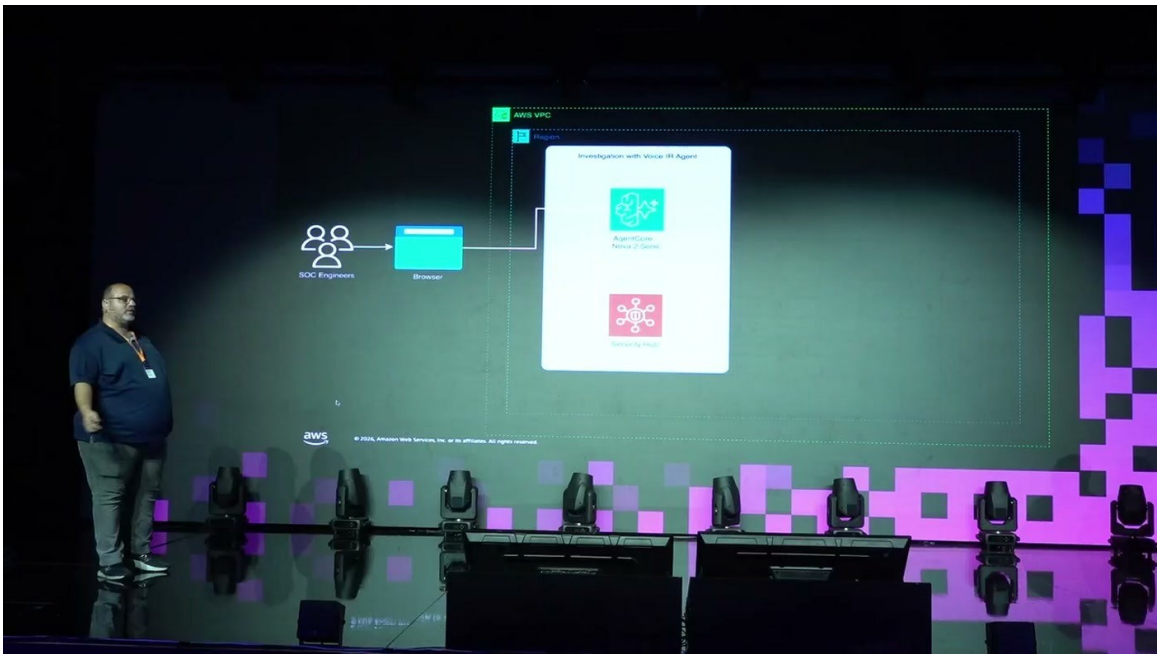
Security Hub כותב event ל-EventBridge; משם הוא מגיע ל-Durable Lambda, שמנהלת את ה-workflow.
הדבר הראשון שהיא עושה הוא לשלוח נוטיפיקציה — בפועל, הודעה בסלאק דרך SNS.



שרשרת הזיהוי: Security Hub (CSPM Detection) ← EventBridge ← Durable Lambda ← SNS להודעה בסלאק.

החקירה

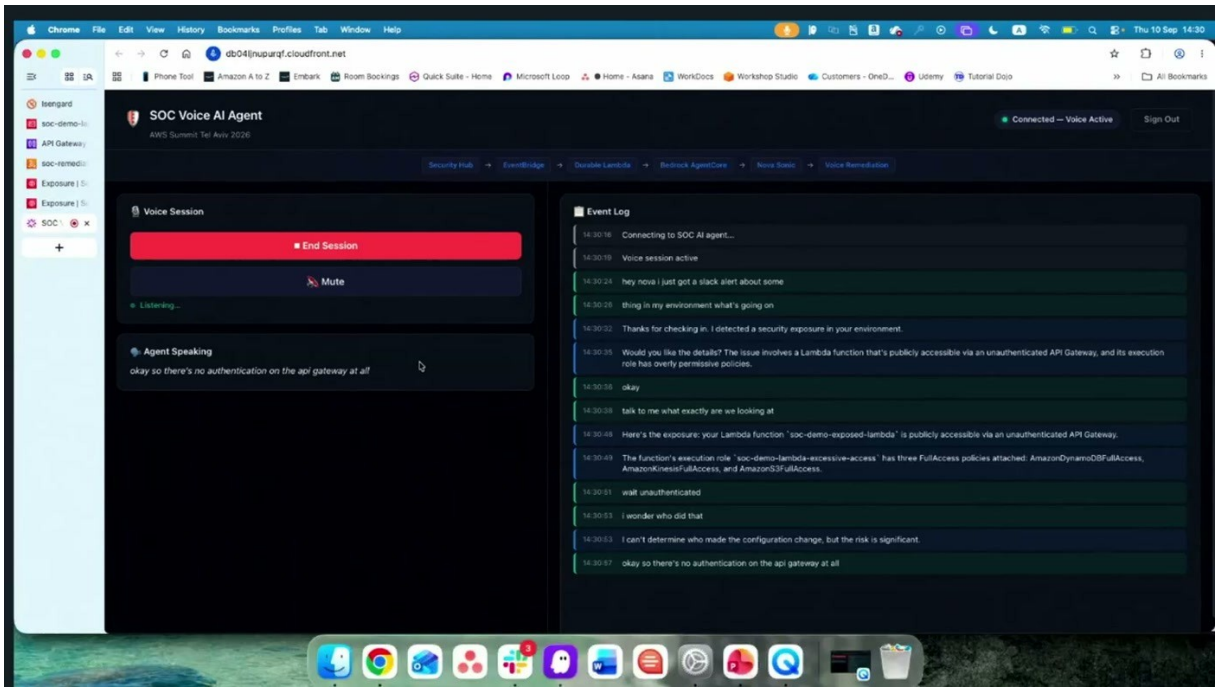
מההודעה בסלאק מגיע מהנדס ה-SOC ל-web application, ושם — במילותיו של שימי — "יש שם איזשהו speech to speech agent עם טולים ויכולות של להסביר מה קרה מה האימפקט, איך כדאי לעשות remediation ולבצע אותו בפועל באמצעות אותה דיורה בלמדה" [7:36].



צד החקירה: מהנדסי SOC ← דפדפן ← הסוכן על AgentCore עם Nova 2 Sonic, שקורא מ-Security Hub.

6. הדמו החי

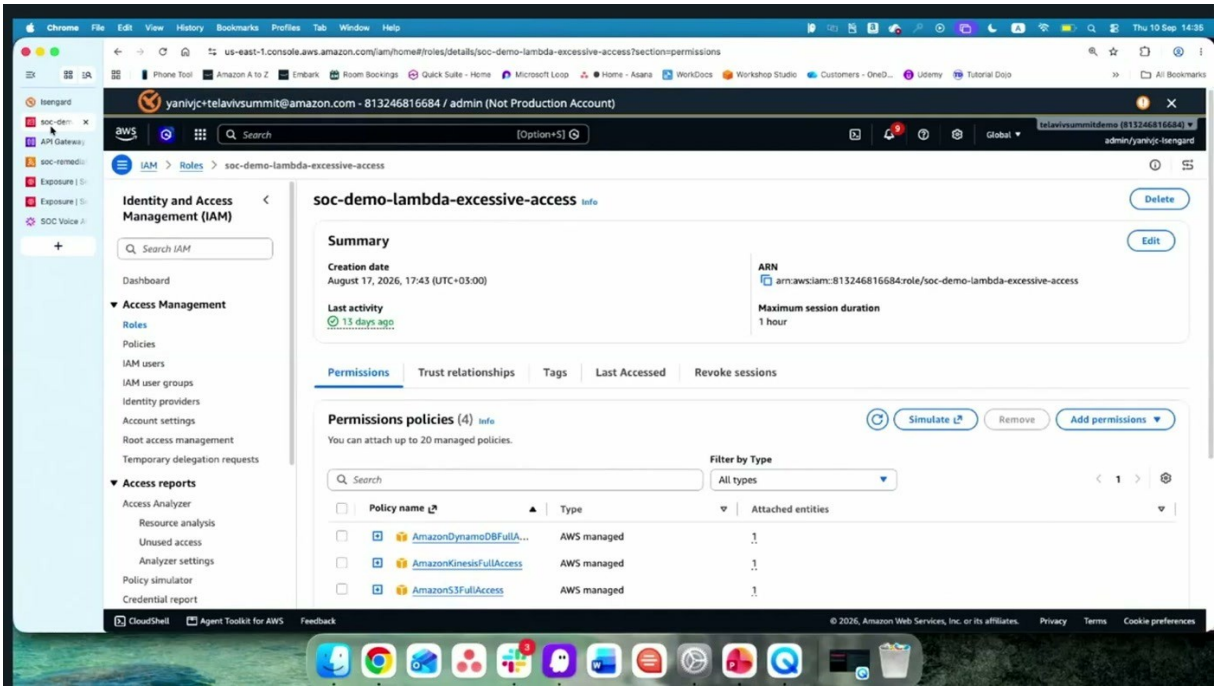
הדמו רץ בשידור חי, על חשבון AWS אמיתי, עם הכרזה מקדימה: "זה כן יהיה Live Voice AI demo as fingers crossed" [7:36]. הרצף היה: התראת Slack בחומרה Medium על API Gateway ללא אותנטיקציה ← הסתכלות ב-Security Hub ← מעבר לאפליקציית הקול.



ממשק ה-SOC Voice AI Agent בזמן שיחה: פאנל הסשן הקולי משמאל, ולוג אירועים מימין שמתעד את הדיאלוג שורה-שורה עם חותמות זמן.

מלוג האירועים במסך עולה מה הסוכן זיהה ומסר: שהפונקציה soc-demo-exposed-lambda נגישה פומבית דרך API Gateway לא מאומת, ושרול ההרצה שלה, soc-demo-lambda-excessive-access, נושא שלוש פוליסות FullAccess. בשיחה עצמה הוא אמר את זה כך [9:10]: "Amazon has 3 full access policies attached. DynamoDB Full Access, Amazon Kinesis Full Access, and Amazon SE Full Access" (השיבוש "SE" בתמלול הוא S3).

> **— יניב קדוש והסוכן, [9:10]** So if someone discovers this endpoint, they can read every S3 bucket, every DynamoDB table, and stream data out of Kinesis? That's basically full data exfiltration. — Exactly.



קונסולת IAM בזמן הדמו: הרול soc-demo-lambda-excessive-access ושלוש פוליסות ה-FullAccess המצורפות לו.

אחרי אישור קולי, הסוכן ביצע את התיקון והודיע על התוצאה [12:25]: "now only has read only access to S3, DynamoDB and CloudWatch logs. The full access policies for Kinesis, DynamoDB and S3 have been removed and a new read only least privilege policy has been applied."

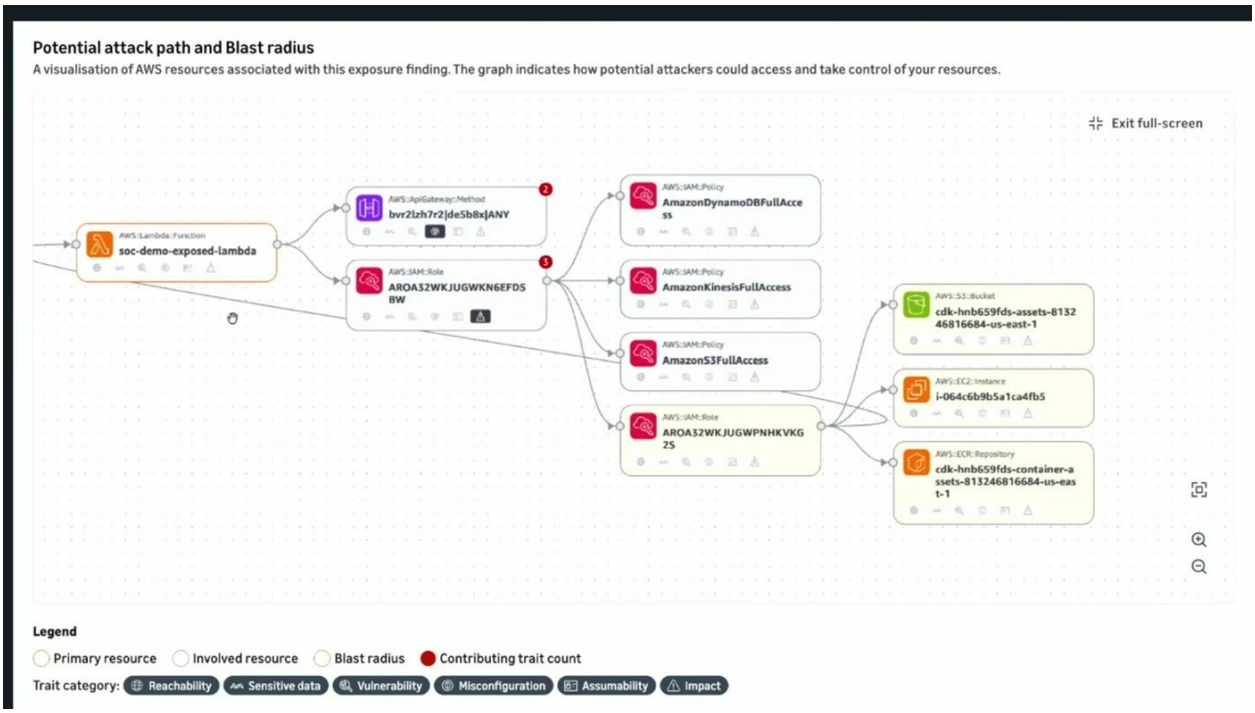
החלק החשוב בדמו היה מה שבא אחריו — האימות הידני. יניב רענן את שני המסכים: "אם אתם זוכרים בהתחלה של הסשן, זה היה authorization none, ויש לנו עכשיו את ה-AWS IAM authorization" [13:56], ואז בלשונית ה-Permissions Policies "עכשיו יש לנו את ה-Least Privilege, Read Only" [13:56]. גם ה-finding ב-Security Hub עבר ל-Resolved.

הערה על אינטראקטיביות בסיכום שלו הדגיש יניב שהשיחה לא הייתה תור-אחר-תור נוקשה: "where I can interrupt the AI agent וגם [23:02] "AI agent interrupts me. יכולת הקטיעה ההדדית (barge-in) היא חלק ממה שהופך את ה-speech-to-speech לשימושי בתרחיש לחץ.

7. Exposure: מה זה בעצם

אחרי הדמו חזר שימי להסביר את הרכיב שרוב הקהל פחות מכיר. עד לפני כשנה, הסביר, Security Hub אסף findings — בעיית קונפיגורציה, ספרייה פגיעה במכונה — ממקורות שלו וממקורות אחרים, והציג אותם. "לפני בערך שנה הוספנו את היכולת" [13:56] "לייצר משהו שנקרא exposure" [15:27] — ממצא מורכב שמתאם בין כמה בעיות ומרכיב מהן סיפור אחד.

במקרה שלנו: API Gateway בלי אותנטיקציה + Lambda עם רול אדמיניסטרטיבי = exposure אחד, שמאפשר לראות attack path ו-impact. שימי תיאר את מה שהגרף מראה: "דרך באמצעות הלמדה הזאת שחשופה דרך API Gateway יש לי גישה לפוליסיס האלה ב-IAM ודרכם גישה לריסורסים האלה בחשבון" [15:27] — ומהגרף הזה גם נגזרת הקריטיות של הממצא.



גרף ה-Potential attack path and Blast radius: מהפונקציה החשופה, דרך המתודה ב-API Gateway ושני רולים, אל שלוש פוליסות FullAccess ומשם אל דלי S3, מופע EC2 ו-repository ב-ECR.

הגרף גם צובע כל צומת לפי trait category — Reachability, Sensitive data, Vulnerability, Misconfiguration, Assumability ו-Impact — כך שאפשר לראות לא רק לאן אפשר להגיע, אלא למה זה מסוכן.

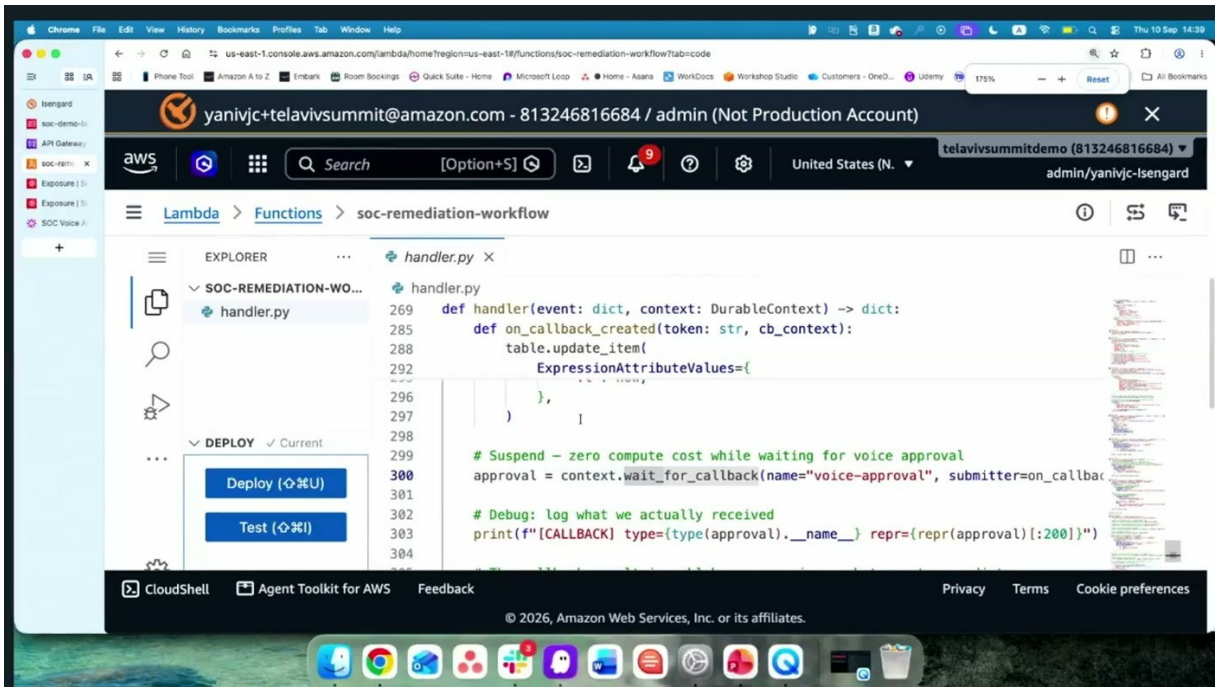
8. Durable Lambda: איך עוצרים לאישור אנושי

זה היה החלק הטכני המרכזי של הסשן. Durable Lambda, הסביר שימי, היא כמו Lambda רגילה — יש לה handler — אלא שה-handler עובד בשלבים, ובשלב מסוים אפשר להגיד לו "תעצור ותחכה שמישהו יגיד לך להמשיך" [16:58].

> **— שימי רוקח, [16:58]** השלב הזה יכול להיות לחכות דקה, עשר דקות, שעתיים שבוע ולכן הוא מעולה ל-human in the loop וורקפלווס וכמובן שבזמן שמחכים לא משלמים על ההרצה של הלמדה

רצף הפעולות שהפונקציה ביצעה בדמו, לפי סדר:

- **פרסור האירוע.** קבלת ה-event מ-EventBridge והבנה מה קרה.
- **יצירת state ב-DynamoDB.** מצב הטיפול בבעיה נשמר בטבלה, "כדי שבעצם יהיה איזשהו סינפון בין האג'נט שיניב דיבר איתו לבין הלמדה הזאת שמנהלת את ה-[16:58] "Walkflow. זה גם המקום שממנו הסוכן שולף בהמשך את ה-callback ID.
- **שליחת נוטיפיקציה ב-SNS.** ההודעה שהופיעה בסלאק בתחילת הדמו.
- **יצירת callback והשהיה.** "זה החלק הייחודי ל-[16:58] "Durable Lambda's — הפונקציה נכנסת ל-wait for callback, עוצרת בשלב הזה, ולא מחויבת על זמן ההמתנה.



השורה שממממשת את ההמתנה בקוד ה-handler, עם ההערה המפורשת: "Suspend — zero compute cost while waiting for voice approval".

מה שנשאר פתוח הוא ה-callback ID: "מחכה עם איזשהו callback ID שמישהו יקרא ל-API של Lambda's ויגיד עכשיו את ה-callback ID הזה אפשר להמשיך, אי אפשר להמשיך ואם המשיכים עם איזה פרמטרים" [16:58]. ברגע שה-callback מתקבל, השלב הבא מבצע את ה-remediation בפועל על בסיס אותם פרמטרים [18:29].

9. הקוד של הסוכן

הסוכן עצמו, הדגיש שימי, "לא מאוד מורכב": סוכן בפייתון על בסיס Strands — ה-framework בקוד פתוח של AWS — במקום LangChain, LangFlow או אחרים. מה שמגדיר אותו הוא system prompt קצר ורשימת טולים.

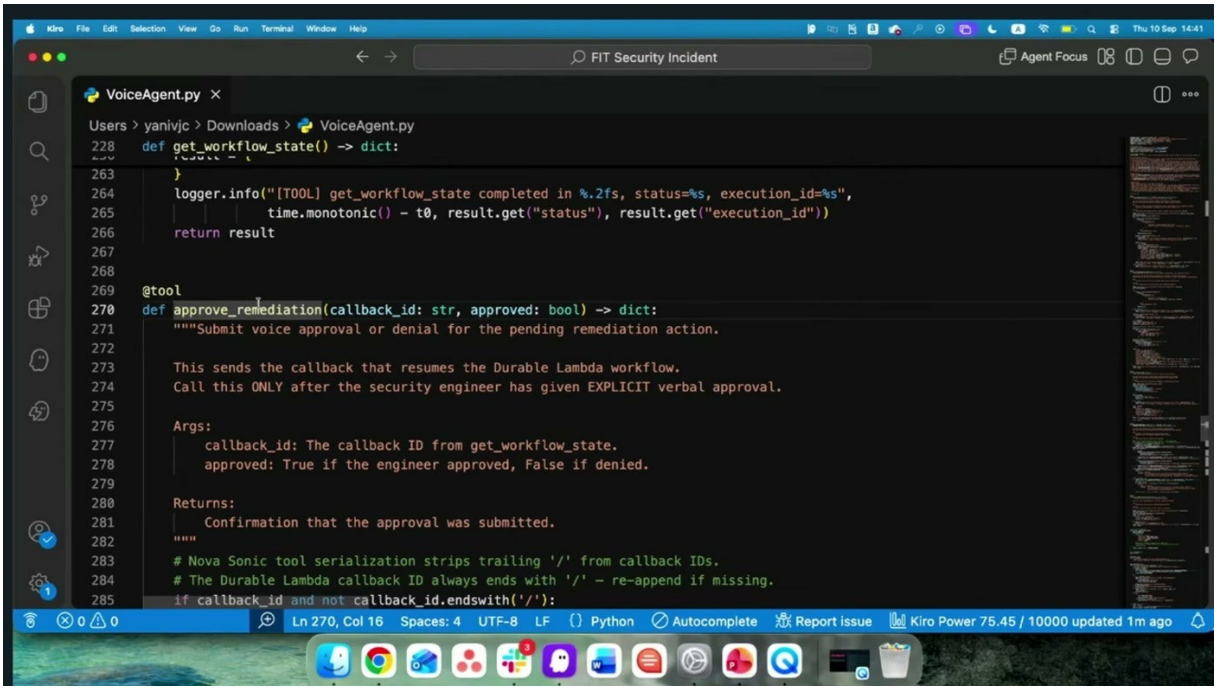
ה-system prompt, בתיאורו: "אתה Senior Sock Engineer אתה עוזר לסוק אנג'ינירס אחרים להבין מה הבעיה שקרתה" [18:29] — להסביר את הבעיה, להביא את ההמלצות של Security Hub, להסביר מה ה-impact, ולבסוף גם להריץ את התיקון, הכול דרך טולים.

list_findings

הטול שמביא מה-DynamoDB את הממצאים האחרונים שעוד לא טופלו. הנימוק העסקי מעניין יותר מהמימוש: "יש צוות של אנשי סוק, יכול להיות שכמה מהם כבר בטיפול, ולכן לא נרצה שטופלו במקביל כמה פעמים" [18:29]. בקוד שהוצג הטול פונה ל-API get_findings_v2 של Security Hub, עם severity כברירת מחדל MEDIUM ועד 20 תוצאות.

approve_remediation

הטול שסוגר את המעגל מול ה-Durable Lambda. כשהמנדס אומר בקול "תריץ את הרמדיאציה", הטול מבצע קריאה ל-API בשם Durable Execution Callback Success עם ה-callback ID שנשלף מ-DynamoDB, ומעביר איתו פרמטרים — בדמו פרמטרים מינימליים, אבל בפרודקשן זה יכול להיות "ID של פלייבוק, פרמטרים ספציפיים, איך לעשות רמדיאשן" [20:00].



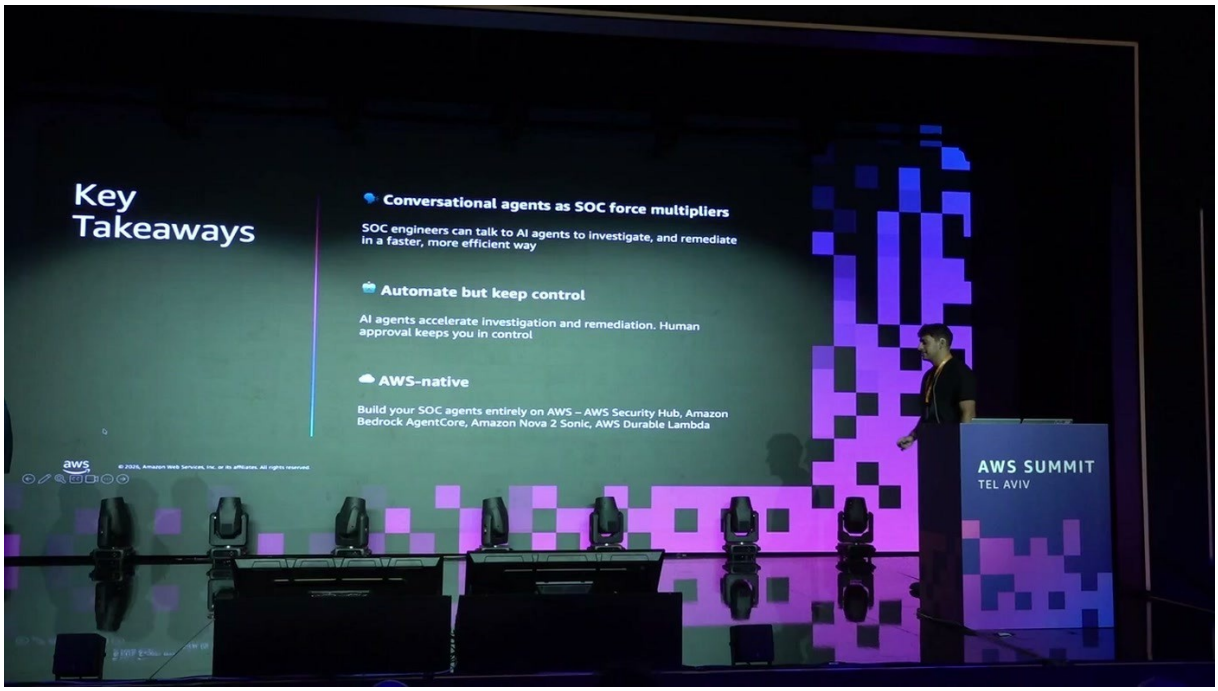
ה-docstring של הטול, שהוא בפועל חלק ממנגנון הבקרה: "Call this ONLY after the security engineer has given EXPLICIT verbal approval."

> **פרט מימוש ששווה לזכור** בהערות בקוד מופיע באג אמיתי שנפתר: סריאליזציית הטולים של Nova Sonic חותכת את ה-/ הסופי מתוך ה-callback ID, וה-ID של Durable Lambda תמיד מסתיים ב-/ — ולכן הקוד מוסיף אותו בחזרה. זה בדיוק סוג החיכוך שמופיע כשמחברים מודל קולי לשירות תשתית.

למה WebSockets

הנקודה האחרונה שהוא הדגיש: "הפרוטוקול שאנחנו עובדים איתו הוא WebSockets, כי אנחנו לא מדברים פה על איזשהו Requesting Response, אלא על Streaming של [20:00] "Voice. זו החלטה שנגררת ישירות מהבחירה ב-speech-to-speech.

10. ה-Key Takeaways של הדוברים



סליד הסיכום עם שלוש הנקודות שהדוברים ביקשו שיישארו.

- **Conversational agents as SOC force multipliers.** מהנדסי SOC יכולים לדבר עם סוכני AI כדי לחקור ולתקן מהר ויעיל יותר, בלי להיות overwhelmed מסביב לשעון.
- **Automate but keep control.** האישור האנושי הוא מה שמשאיר את השליטה בידיים. יניב חזר על זה כנקודה עצמאית, לא כהערת שוליים.
- **AWS-native.** אפשר לבנות את הסוכן כולו על AWS: Security Hub, Bedrock AgentCore, Nova 2 Sonic ו-Durable Lambda.

ובנימה זהירה יותר, שראוי לצטט במלואה: "זה גם חשוב לדעת זה לא One Way או My Way אתם יכולים לעשות את זה ב-Alternative Methods אבל זה כן אחלה use case להשתמש ב-[21:31] "Voice AI.

11. מה לוקחים מכאן

- **הרעיון שראוי להעתיק הוא לא הקול — אלא ה-callback.** התבנית של workflow שנעצר ב-wait_for_callback, לא צורך משאבים בזמן ההמתנה, וממשיך רק כשגורם חיצוני מאשר עם פרמטרים — רלוונטית לכל תהליך רגיש שמערב סוכן, גם בלי שמץ של Voice AI.
- **Exposure שווה בדיקה בפני עצמו.** גם ארגון שלא ייגע בסוכנים קוליים יכול להפיק ערך מיידי מיכולת ה-exposure של Security Hub: מעבר מ-findings בודדים לשרשרת מתואמת עם attack path ו-blast radius מוריד רעש ומתעדף אמיתי.
- **הדמו הוא Level 300 אבל התרחיש מפושט.** החשיפה נבנתה במכוון, התיקון היה יחיד ומוגדר מראש, וה-playbook היה קבוע. הפער מפרודקשן — ריבוי ממצאים, playbooks מרובים, הרשאות הסוכן עצמו — לא נדון בשן.
- **ההרשאות של הסוכן הן השאלה הפתוחה.** סוכן שיכול להסיר ולהחליף פוליסות IAM הוא בעצמו נכס בעל הרשאות גבוהות. הבקרה שהוצגה היא האישור הקולי; בקרות נוספות (הגבלת ה-playbooks, הפרדת חשבונות, תיעוד) לא הוצגו.
- **עברית איננה בתמונה כרגע.** לארגון ישראלי שבו שפת העבודה של צוות ה-SOC היא עברית, זו מגבלה מעשית. יניב ציין עברית כ-roadmap ולא כזמינות, והדגיש את הנקודה בבדיחה: "אני יודע נובה לא מדברת בעברית" [13:56].
- **הבנייה זמינה מיידית.** כל ארבעת השירותים הם GA או הוכרזו ב-re:Invent 2025, והדוברים הזמינו במפורש להתחיל לבנות "מחר" [21:31]. POC בהיקף של תרחיש אחד נראה סביר.

נספח: מילון מונחים

מונח	מה זה בסשן הזה
Amazon Nova 2 Sonic	מודל speech-to-speech של AWS. הסוכן הקולי בדמו מבוסס עליו. תומך באנגלית, צרפתית, ספרדית ובמבטאים שונים; עברית הוזכרה כ-roadmap.
cascaded מול speech-to-speech	cascade = תמלול לטקסט, מודל טקסט, סינתזה חזרה לקול. speech-to-speech מדלג על השלבים ומקצר latency.
Bedrock AgentCore Runtime	סביבת הרצה מנוהלת לסוכנים בקונטיינר, כולל gateway, authorization, observability, identity ו-capacity.
Strands	ה-framework בקוד פתוח של AWS לבניית סוכנים בפייתון. בתמלול האוטומטי הופיע כ-"Strengths".
Lambda Durable Functions	Lambda רב-שלבית, stateful, שיכולה לרוץ מעל 15 דקות ולהשהות את עצמה. הוכרזה ב-re:Invent 2025.
wait_for_callback	הקריאה שמשהה את ה-Durable Lambda עד שגורם חיצוני יקרא ל-API עם ה-callback ID. אין חיוב compute בזמן ההמתנה.
Security Hub Finding	ממצא בודד: קונפיגורציה שגויה, ספרייה פגיעה וכדומה.
Security Hub Exposure	ממצא מורכב שמתאם בין כמה findings ומציג attack path ו-blast radius. נוסף לשירות כשנה לפני הסשן.
Blast radius	מפת המשאבים שתוקף יוכל להגיע אליהם דרך החשיפה — בדמו: S3, DynamoDB, Kinesis, EC2 ו-ECR.
Human in the loop	עצירה מובנית בתהליך לאישור אנושי. כאן — אישור קולי מפורש לפני ביצוע ה-remediation.
CSPM	Cloud Security Posture Management — קטגוריית הזיהוי שממנה מגיע ה-finding בסליד הארכיטקטורה.