interlynk

Checklist voor SBOM-beheer in de onderneming

Een praktische maatregelenchecklist voor het beheren van SBOM's op schaal, opgebouwd rond de vijf plekken waar SBOM-beheer in de onderneming vastloopt en de maatregelen die elk daarvan onder controle houden.

SBOM's genereren is het makkelijke deel. Duizenden ervan beheren over honderden teams, leveranciers en overnames heen is het deel waar programma's stuklopen. Werk dit door als een volwassenheidscheck. **Als je een punt niet kunt afvinken, is dat een gat dat het waard is te dichten voordat de volgende zeroday het voor je vindt.**

1 SBOM-wildgroei beheersen (volume)

De map met bestanden en de spreadsheet houden lang op te werken voordat ze een fout geven. Bepaal het opslagmodel voordat je tegen de muur loopt.

- [ ] SBOM's worden als stap in de build-pipeline naar een systeem van registratie geschreven, niet met de hand geüpload.
- [ ] Elke SBOM is gekoppeld aan een onveranderlijke build- of release-identificatie (zoals de artefact-digest), zodat de juiste jaren later terug te vinden is.
- [ ] De opslag is geïndexeerd op component, zodat een lookup een query is en geen scan door miljoenen bestanden.
- [ ] Een bewaarbeleid houdt SBOM's voor uitgeleverde en vrijgegeven artefacten aan om aan de wettelijke bewaarplicht te voldoen, en ruimt wegwerp-CI-builds op.
- [ ] Je kunt bij benadering aangeven hoeveel SBOM's je hebt en waar ze staan.

2 Elke SBOM op één lijn brengen (formaat en identiteit)

Je kunt geen bestanden samenvoegen die het oneens zijn over structuur, en geen componenten dedupliceren die het oneens zijn over identiteit. Breng alles in vorm voordat je het opslaat.

- [ ] Binnenkomende SBOM's worden bij ingestie omgezet naar één canoniek intern formaat en één canonieke versie, waarbij het originele bestand voor de audit intact blijft.
- [ ] Generatoren zijn ingesteld om purl als primaire componentidentificatie uit te geven, met CPE als terugval.
- [ ] Componenten die alleen als vrije tekst binnenkomen, worden gemarkeerd in plaats van stilzwijgend geaccepteerd.
- [ ] Een resolver koppelt aliassen, pakketcoördinaten en CPE's aan één canonieke sleutel, ondersteund door een aliastabel die je onderhoudt.
- [ ] Elke SBOM wordt bij ingestie gevalideerd tegen het specificatieschema, en misvormde bestanden worden bij de deur geweigerd.
- [ ] De toolchains van overgenomen bedrijven lopen via hetzelfde ingestie- en normalisatiepad.

3 SBOM's levend houden (veroudering)

Een opgeslagen SBOM is een momentopname die veroudert. Monitoring maakt er weer een actueel antwoord van.

- [ ] Genereren en beoordelen zijn ontkoppeld: SBOM's worden op build-moment gegenereerd en continu opnieuw beoordeeld tegen verse kwetsbaarheidsdata.
- [ ] Een nieuwe CVE wordt automatisch gekoppeld aan de precies getroffen producten en versies.
- [ ] Bevindingen gaan zonder handmatige triage naar het team dat het betreffende product bezit.
- [ ] VEX wordt toegepast zodat de herbeoordeling signaal oplevert, en VEX-verklaringen blijven bestaan zodat een eenmaal gemaakte onderdrukking latere herscans overleeft.
- [ ] Monitoring weerspiegelt wat daadwerkelijk in gebruik of op de markt is, niet alleen wat gebouwd werd.

4 Leveranciers aan een standaard houden (SBOM's van derden)

De inventaris van derden verbergt de gevaarlijkste onbekenden en is meestal het slechtst beheerde deel. Leg de kwaliteitslat vast in het contract, zolang je nog invloed hebt.

- [ ] SBOM-levering is een contractuele verplichting die formaat, versie en een minimumset vereiste elementen vastlegt.
- [ ] Elke leverancier-SBOM wordt bij ontvangst gevalideerd en beoordeeld op volledigheid en kwaliteit van de identificatoren.
- [ ] SBOM's die tekortschieten, worden gemarkeerd of geweigerd voordat ze in het systeem van registratie komen.
- [ ] Leverancier-SBOM's worden genormaliseerd naar dezelfde opslag en dezelfde continue monitoring als je eigen uitvoer.
- [ ] Een aangewezen eigenaar (meestal inkoop samen met security) is verantwoordelijk voor het nabellen bij onvolledige of ontbrekende leverancier-SBOM's.

5 Geef het programma een eigenaar (governance)

Onder elk technisch falen hierboven ligt een organisatorisch falen. Verspreide verantwoordelijkheid is waarom de portfoliobrede vraag onbeantwoord blijft.

- [ ] Er is één aanspreekbare eigenaar voor het programma, doorgaans een lead voor toeleveringsketen- of productsecurity.
- [ ] Een RACI maakt de verdeling expliciet: engineering genereert, een centrale security- of platformfunctie beheert het systeem van registratie, compliance consumeert.
- [ ] Beleid wordt centraal bepaald (formaat, identificatoren, bewaring, leverancierseisen) en lokaal per team uitgevoerd.
- [ ] De gemiddelde tijd tot een antwoord op „waar zijn we kwetsbaar” wordt als metric bijgehouden.
- [ ] De zerodayvraag wordt als geplande oefening doorlopen, voordat een aanvaller haar voor je doorloopt.

Hoe het eruitziet als het werkt

Als deze maatregelen op hun plek staan, werken vijf capaciteiten samen als één systeem. Elke SBOM, intern en van leveranciers, staat in één systeem van registratie, ongeacht formaat of versie. De componentidentiteit is genormaliseerd, zodat een portfoliobrede vraag één antwoord oplevert in plaats van een verzoeningsproject. Opgeslagen SBOM's worden continu opnieuw beoordeeld, en een nieuw gemelde CVE brengt elk getroffen product vanzelf naar boven. Leverancier-SBOM's nemen dezelfde horde als je eigen SBOM's voordat ze worden geaccepteerd. De inventaris voedt de securitystack en stuurt kwetsbaarhedenbeheer, VEX en incident response aan.

Een onderneming kan een miljoen SBOM's hebben en er geen enkele beheren. De alinea hierboven beschrijft hoe beheren er werkelijk uitziet.
