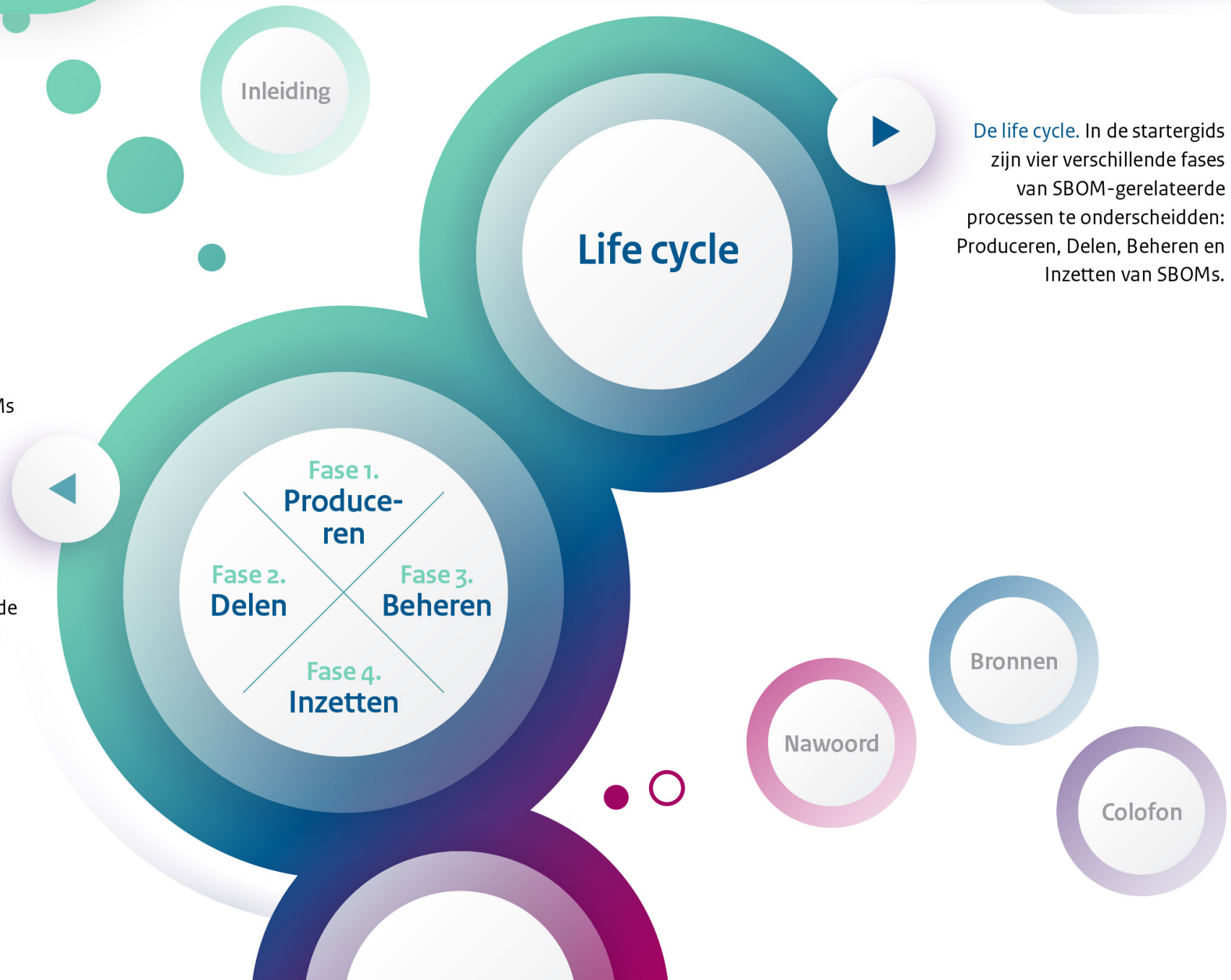
Nationaal Cyber Security Centrum
Ministerie van Justitie en Veiligheid

Software Bill of Materials (SBOM)

Een betrouwbare life cycle

TNO innovation for life

Gebruik dit figuur om door de gids te bladeren



De life cycle. In de startergids zijn vier verschillende fases van SBOM-gerelateerde processen te onderscheidden: Produceren, Delen, Beheren en Inzetten van SBOMs.

Fases: Produceren, Delen, Beheren, Inzetten. Om de functionele doelen en security doelen voor SBOMs inzichtelijk te maken, worden deze per fase van de SBOMs life cycle geïdentificeerd. Na beschrijving van de doelen identificeren we de belangrijkste valkuilen en de mogelijkheden om deze te voorkomen.

Inleiding

Terug naar startfiguur

Introductie

In het recente verleden hebben zich al verscheidende *supply chain* incidenten voorgedaan, waarbij organisaties zijn blootgesteld aan cyberrisico's doordat ze gebruikmaakten van software die door anderen was ontwikkeld. Hierbij ging het soms om volledige softwareproducten, maar in andere gevallen betrof het extern ontwikkelde *dependencies* die in eigen software werden gebruikt. Bekende voorbeelden hiervan zijn Log4j[1] en de backdoor in de populaire xz tool in 2024[2]. ENISA heeft gecompromitteerde software supply chains geïdentificeerd als de meest prangende cyberdreiging voor het jaar 2030 [1↗]. Deze dingen tonen aan hoe belangrijk het is om inzicht te hebben in de samenstelling, afhankelijkheden, totstandkoming en inzet van software binnen de eigen organisatie. Het juiste gebruik van Software Bill-of-Materials (SBOM) zal een essentieel onderdeel vormen om dit inzicht te realiseren.

Eerder brachten TNO en het NCSC de SBOM Startergids [2↗] uit. Hierin werd het *wat, waarom,* en *hoe* van de SBOM beschreven. De startergids legt uit wat SBOMs zijn, waarom de SBOM nuttig kan zijn voor een organisatie, en hoe de inzet van SBOM met de juiste processen en afspraken kan worden gerealiseerd. Deze nieuwe gids bouwt voort op de startersgids en geeft een gedetailleerd en technisch overzicht van de *life cycle* van de SBOM. De focus ligt hierbij op hoe deze op een veilige en betrouwbare manier door organisaties kan worden ingericht. De term "SBOM life cycle" wordt gebruikt om alle processen aan te duiden die zorgen voor de productie van SBOMs, het delen ervan met andere partijen, beheer en archivering van de SBOM-informatie, en het beschikbaar stellen van deze informatie voor verdere cyberveiligheidsprocessen. Het opzetten van een veilige en betrouwbare life cycle is daarmee dus een organisatie-overstijgende aangelegenheid. In deze gids worden de fases in de life cycle verder toegelicht en worden functionaliteits- en security-doelen per fase besproken. Hierbij wordt gekeken naar hoe SBOM-gerelateerde tooling deze functionaliteits- en security-doelen kan vervullen, welke praktische risico's bestaan en hoe deze gemitigeerd kunnen worden.

1 Log4j: Wat hebben we tot nu toe geleerd? | Expertblogs | Nationaal Cyber Security Centrum (ncsc.nl)
2 The XZ-factor: social vulnerabilities in open source projects | By our experts | National Cyber Security Centre

Inleiding

Terug naar startfiguur

Het belang van SBOMs

Het binnenhalen van complete en accurate SBOMs voor de software die een organisatie gebruikt, is slechts een eerste stap. Een goed ingerichte SBOM life cycle dient onderdeel te zijn van een bredere aanpak op de (deels overlappende) gebieden van:

- **cybersecurity supply chain risicomanagement:** een breed proces waarbij mogelijke risico's die voorvloeien uit de complexiteit van IT- en OT-supply chains worden geïdentificeerd, beoordeeld en gemitigeerd.
- **vulnerability & exposure management:** het detecteren en beheren van kwetsbaarheden in het IT- en OT-landschap van de eigen organisatie en het analyseren en verkleinen van mogelijke aanvalsoppervlakken.

Door de afhankelijkheden van software met behulp van SBOMs vast te leggen, krijg je inzicht in (indirecte) afhankelijkheid op oudere, niet langer ondersteunde, of kwetsbare dependencies. Een organisatie kan op hand van deze informatie druk uitoefenen op hun leveranciers voor updates, kiezen voor alternatieve producten, of haar infrastructuur anders gaan inrichten om zo veiligheidsproblemen te voorkomen. Deze kennis kan ook worden ingezet om de gevolgen van nieuw gerapporteerde kwetsbaarheden voor een organisatie te bepalen of bij het kiezen van mitigerende acties.

Compliance

Een andere belangrijke reden om SBOMs in te zetten binnen een organisatie is *compliance management*. Wetgeving betreft steeds vaker ook het digitale domein. In 2025 treedt de Cyber Resilience Act [3↗] van de EU in werking. Leveranciers van hardware en software worden verplicht om hun klanten te informeren over de samenstelling en eventuele kwetsbaarheden in de door hun geleverde producten. Zij dragen zorg voor de cyberveiligheid van deze producten tijdens de hele levenscyclus. SBOMs kunnen worden gebruikt door leveranciers en gebruikers om ook aantoonbaar en transparant te voldoen aan deze regelgeving.

Deze gids is een handreiking voor organisaties voor het inrichten van hun eigen SBOM-life cycle. De voorgestelde functionaliteits- en securitydoelen dienen om organisaties te helpen bewuste keuzes te maken bij het aanschaffen, bouwen, of uitbesteden van SBOM gerelateerde tooling en processen, om zo een veilige en betrouwbare SBOM life cycle te realiseren. De informatie in dit document is gebaseerd op deskresearch naar huidige best practices, bestaande SBOM tooling, data- en risicomanagement-standaarden, en interviews met experts van het NCSC.

Inleiding

Terug naar startfiguur

Securitydoelen

Het belangrijkste doel bij het gebruik van SBOM is ervoor te zorgen dat accurate en complete informatie over de samenstelling van software beschikbaar is voor andere (cybersecurity) processen. Als deze informatie niet beschikbaar, incompleet of onbetrouwbaar is, kan dit nadelige gevolgen hebben. Een organisatie kan op basis van een onnauwkeurig overzicht aannemen dat ze niet gevoelig is voor een nieuwe kwetsbaarheid. Dit leidt tot een *false sense of security*. Het voornaamste doel van een juist ingerichte SBOM life cycle is om dit soort situaties te voorkomen. Hiernaast is het belangrijk dat er bij het inrichten van de SBOM life cycle bepaalde securitydoelen worden gewaarborgd.

In deze gids zullen wij verder gebruikmaken van het STRIDE-model [4↗] om per life cycle fase deze securitydoelen en aanverwante dreigingen te duiden. Het STRIDE-model verdeeld dreigingen in zes verschillende categorieën. Elk van deze dreigingen heeft een corresponderend mitigerend securitydoel. Deze worden gegeven door de welbekende CIA-triad (Confidentiality, Integrity, Availability) met de aanvulling van Authenticity, Authorization en Non-repudiation (AAN). De dreigingen, samen met de corresponderende securitydoelen en voorbeelden, worden beschreven in **Tabel 1**.

Tabel 1: STRIDE dreigingen en CIA+AAN ter mitigatie.

Dreiging	Securitydoel	Voorbeeld dreiging
Spoofing	Authenticity	Een aanvaller doet zich voor als een vertrouwde leverancier.
Tampering	Integrity	Een aanvaller past de inhoud van een door een leverancier geleverde SBOM aan.
Repudiation	Non-repudiation	Een SBOM wordt gegeneerd zonder dat adequaat wordt vastgelegd door wie en op wat voor manier deze tot stand is gekomen.
Information disclosure	Confidentiality	Een aanvaller kan de SBOM lezen zonder dat hij de bedoelde ontvanger is.
Denial of service	Availability	Een organisatie heeft vanwege een aanval geen toegang meer tot haar SBOM-opslag.
Elevation of privilege	Authorization	Een aanvaller of ongeautoriseerde medewerker is in staat om SBOMs te laten genereren of publiceren door de CI-pipeline van een leverancier.

Leeswijzer →

Inleiding

Terug naar startfiguur

Leeswijzer

Deze gids bouwt voort op informatie in de SBOM Startergids [2↗]. Voor een uitgebreide uitleg van SBOM en de wijze waarop deze nuttig kunnen zijn voor een organisatie is het raadzaam om eerst de startergids te lezen. In het volgende hoofdstuk *De life cycle* worden de vier onderscheiden fases van de SBOM life cycle beschreven. Hierna volgt het hoofdstuk *Fases*, waarin functionaliteits- en securitydoelen worden gegeven voor het inrichten van de verschillende processen in de SBOM life cycle.

De life cycle →

De life cycle

Terug naar startfiguur

De vier fases

Zoals eerder genoemd, zal een organisatie SBOMs willen verzamelen en beschikbaar maken voor vulnerability en compliance management processen. Om dit doel te realiseren moeten een aantal processen zijn ingericht. Samen noemen we deze processen de SBOM life cycle. In de startergids zijn vier verschillende fases van SBOM-gerelateerde processen onderscheidden: (1) **Produceren**, (2) **Delen**, (3) **Beheren**, en (4) **Inzetten** van SBOMs.

Fase 1: Produceren ↗

Produceren betreft het zelf opstellen van een SBOM voor een gegeven softwareproduct. Dit kan handmatig en ad-hoc plaatsvinden, of beter nog als onderdeel van een geïntegreerd software ontwikkelproces [5↗]. Externe en zelfgeproduceerde SBOMs kunnen vervolgens door een organisatie ingenomen en verder verwerkt worden.

Fase 2: Delen ↗

Het delen van een SBOM is het tweede segment van de life cycle, waarbij de geproduceerde SBOM, al dan niet samen met het onderliggende softwareproduct, door de leverancier ter beschikking wordt gesteld aan de afnemer, zodat deze vervolgens de SBOM kan overbrengen naar zijn omgeving [6↗].

Fase 3: Beheren ↗

Na de inname van een SBOM, is deze nu aanwezig in de SBOM-beheeromgeving van de organisatie. Hier kunnen voorbereidende bewerkingen op de SBOM worden uitgevoerd, zoals het controleren van de integriteit van de SBOM. De SBOMs kunnen hierna gelinkt worden aan kwetsbaarheidsinformatie en securityadviezen en verrijkt worden met informatie uit andere bronnen.

Fase 4: Inzetten ↗

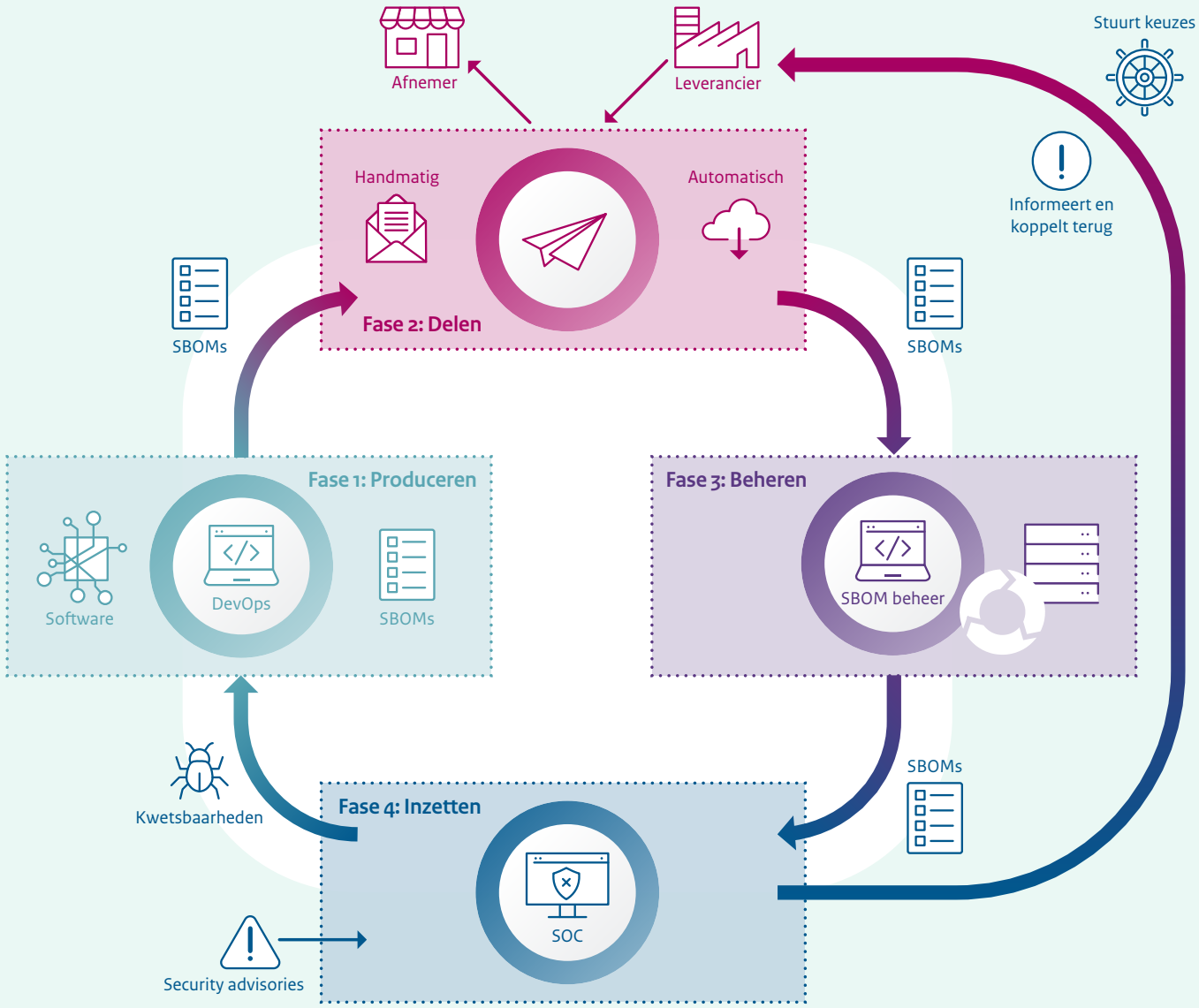
Tot slot is tijdens het inzetten waar de SBOM tot zijn recht komt: als deel van de risk management, exposure management en compliance management processen.

Figuur 1 geeft een schematische weergave van deze fases binnen een organisatie en in relatie met andere partijen.

Doelen →

Kader sluiten

Figuur 1 - Overzicht van SBOM processen binnen de organisatie



Terug naar startfiguur

De life cyc

De vier fases

Doelen

dat deze vervol-
geving [6↗].

vezig in de SBOM-
nen voorberei-
evoerd, zoals het
e SBOMs kunnen
ormatie en
matie uit andere

tot zijn recht komt:
anagement

van deze fases
re partijen.

De life cycle

Terug naar startfiguur

Doelen

Om de functionele doelen en security doelen voor SBOMs inzichtelijk te maken, worden deze in dit hoofdstuk per fase van de SBOMs life cycle geïdentificeerd. De functionele doelen zijn de functionaliteiten die een organisatie zou moeten realiseren bij het inrichten van hun eigen SBOM life cycle management processen. De securitydoelen dienen om security dreigingen die in elke fase voor kunnen komen te mitigeren. Na de beschrijving van de doelen identificeren we de belangrijkste valkuilen voor de fase en de mogelijkheden om deze te voorkomen met behulp van de juiste tooling en processen.

Belangrijke stappen vooraf

Bij ieder fase geven we naast de te behalen doelen ook concrete stappen die een organisatie kan zetten bij het inrichten van het SBOM life cycle management proces. Om goed te beginnen met het implementeren van dit proces, kan je de volgende voorbereidende stappen nemen:

1. Bepaal wie de stakeholders zijn voor het implementeren en integreren van de SBOM life cycle.
2. Bepaal wie toegang moet hebben tot SBOM-informatie.
3. Ga in gesprek met je leveranciers:
   a. Zijn zij al bezig met SBOM?
   b. Wanneer denken ze in staat te zijn SBOMs te leveren?
4. Zoek andere partijen, bijvoorbeeld uit dezelfde sector, om van te leren en ervaringen uit te wisselen.
5. Blijf op de hoogte van nieuwe adviezen en technologische ontwikkelingen.

Fase 1: Produceren →

Produceren

Fase 1.

Terug naar startfiguur

Produceren

Deze eerste stap van de life cycle beslaat het samenstellingsproces van een SBOM. Met behulp van tools worden SBOMs gegenereerd voor specifieke software artefacten, zoals binaries en container-images. Deze tools verzamelen en registreren de specifiek deelcomponenten, directe en indirecte dependencies, runtime dependencies, hun versienummers en andere herkomst informatie. De benodigde informatie over deze afhankelijkheden kan via twee wegen aangevoerd worden: door zelf het overzicht te genereren aan de hand van source code of een artefact, of door het integreren en combineren van eerder samengestelde SBOMs. De componenten van een softwareproduct worden beschreven volgens een gekozen SBOM-standaard. Binnen de productiestap van de life cycle is één functioneel doel geïdentificeerd:

Tabel 2: functionele doelen voor de produceerfase.

Functioneel doel	Beschrijving
Creation	Het genereren van een SBOM voor een specifiek software artefact.

Het creëren van correcte SBOMs is het fundament waarop de verdere life cycle is gebaseerd.

Naast functionele doelen zijn er ook security doelen waaraan de produceerstap moet voldoen. De relevante eisen, gecategoriseerd volgens het STRIDE-model uit sectie 1.2, zijn als volgt:

Tabel 3: security doelen voor de produceerfase.

Security doel	Beschrijving
Authenticity	Het moet zeker zijn dat de informatie waaruit de SBOM gemaakt wordt van een betrouwbare bron komt.
Integrity	Alle gegevens die deel uitmaken van het productieproces moeten onaangetast op worden genomen in de SBOM.
Non-repudiation	De bron van de SBOM kan met zekerheid worden vastgelegd en de ontvanger heeft bewijs van de identiteit van de bron.
Confidentiality	Een aanvaller kan de SBOM niet lezen, deze is versleuteld of alleen ter beschikking gesteld aan geautoriseerde partijen.
Authorization	Een SBOM mag alleen bewerkt worden door processen of gebruikers met de juiste autorisatie daarvoor.

Produceren

Fase 1.

Terug naar startfiguur

Met de huidige tooling is het mogelijk om deze security doelen grotendeels af te dekken. Het probleem is het gebrek aan (de adoptie van) standaarden. Standaardisatie en inter-operabiliteit tussen de verschillende standaarden van SBOMs is nog volop in ontwikkeling.

- **Authenticity** en **Integrity** kunnen worden vervuld door cryptografische signatures en integriteitschecks op de *input* voor de SBOM, zoals source code, binaries, en images. Dit komt neer op het goed inrichten van de software supply chain door de producent.
- **Authorization** kan worden gewaarborgd door de juiste beveiliging van het softwareontwikkelproces en de bouwomgeving.
- **Confidentiality** kan worden gewaarborgd door middel van versleuteling van SBOMs of door alleen (geautomatiseerd) SBOMs te delen met geautoriseerde partijen.
- **Non-repudiation**. Organisaties moeten er vertrouwen kunnen hebben dat de SBOMs die zij ontvangen verifieerbaar door de juiste bron zijn verzonden. Door middel van attestaties kan dit worden vastgelegd door een producent, en achteraf worden gecontroleerd.

Produceren

Fase 1.

Terug naar startfiguur

Waar loop je tegenaan?

Tijdens het produceren zijn er een aantal risico's waar de leverancier tegenaan kan lopen. Het meest prominente probleem betreft de mogelijkheid van het produceren van incomplete SBOMs. Om SBOMs in te zetten voor vulnerability management, is het van essentieel belang dat ze compleet en accuraat zijn[3]. Dit houdt in dat alle softwarecomponenten en afhankelijkheden (tot een bepaalde diepgang) bekend zijn en correct geregistreerd. Wanneer dit echter niet gebeurt, kan het voorkomen dat belangrijke kwetsbaarheden niet gedetecteerd worden, een false negative, waardoor de gebruikers van het softwareproduct ten onrechte denken niet kwetsbaar te zijn.

Op dit moment zijn er zorgen over de volwassenheid van beschikbare tooling en de kwaliteit van de geproduceerde SBOMs. Sommige populaire tools blijken niet in staat om alle belangrijke dependencies vast te leggen [7↗], bijvoorbeeld omdat een dependency niet op de gebruikelijke manier is geïnstalleerd. Verschillende tools zullen ook nog eens (subtiel) verschillende SBOMs genereren.

Het is ook mogelijk dat opzettelijk informatie wordt achtergehouden door een leverancier, bijvoorbeeld omdat het nog niet vanzelfsprekend is dat leveranciers een volledig overzicht creëren van hun afhankelijkheden. Dit zou bijvoorbeeld kunnen gebeuren doordat leveranciers de noodzaak van SBOMs nog niet of onvoldoende inzien om complete SBOMs te produceren [8↗].

Supply chain security

Daarnaast zouden malafide actoren ook mogelijk toegang kunnen verkrijgen tot de SBOM-creërende processen van een leverancier, bijvoorbeeld via 'traditionele' cyberaanvallen of inside actors. Door ongeautoriseerde wijzigingen aan te brengen in dependencies, zoals het vervangen van een dependency met een met malware geïnfecteerde versie, zonder dat dit in de SBOM zichtbaar is, kan er bij afnemers een false sense of security worden veroorzaakt.

De maker van een SBOM moet dus zorgen dat de beveiliging van de eigen software supply chain, ontwikkelomgeving, en software bouwomgeving goed op orde is. De correctheid van de SBOM is hiermee een recursief en mogelijk partij-overstijgend probleem. Niet alleen de integriteit van de directe dependencies is immers belangrijk, maar ook die van de dependencies van de directe dependencies, de bouwomgeving van de dependencies, en zo verder.

Hoe ga je hiermee om →

3 Lees ook het verdiepend kader "Kwaliteit van SBOMs" in de startergids

Produceren

Fase 1.

Terug naar startfiguur

Hoe ga je hiermee om?

Om de correctheid van een SBOM voldoende te kunnen waarborgen, zal in eerste instantie de integriteit van het onderliggende softwareproduct, waarvoor de SBOM wordt samengesteld, moeten kunnen worden gegarandeerd. Het inrichten van een volledig veilig maakproces en software supply chain valt buiten de scope van deze gids. De benodigde beveiligingsmaatregelen zullen afhangen van het risicoprofiel van zowel de leverancier als de afnemer. Recent zijn er verschillende standaarden opgekomen voor de beveiliging van software supply chains, zoals SLSA [9↗] en de OWASP SCVS [10↗]. De laatste heeft zelfs specifieke requirements voor SBOMs. Het maken, delen en controleren van attestaties over de herkomst en totstandkoming van de softwareproducten en SBOMs zal een belangrijk onderdeel vormen van de oplossing.

Deze attestaties zijn cryptografisch gesigneerde verklaringen waarmee een producent aan kan geven dat ze voldoen aan een vooraf opgestelde vereisten, zoals dat alle software gebouwd is in een beveiligde omgeving, volgens een vooraf vastgelegd proces en door geautoriseerde personen. Tooling als in-toto [11↗] en sigstore [12↗] kan hiervoor worden ingezet en wordt ook al ondersteund door steeds[4] meer[5] open-source projecten voor het toevoegen van provenance informatie. Bij het kiezen voor bepaalde dependencies boven alternatieven kan je dus kijken in welke mate deze hun supply chain veiligheid en kwaliteitscontroles op orde hebben[6]. Een andere manier voor het verkleinen van supply chain risico's is het reduceren van externe dependencies.

Daarnaast is het bij de inrichting van het eigen SBOM-productieproces belangrijk om verschillende tools met elkaar te vergelijken, om te zien of deze complete SBOMs genereren. Dit geldt in het bijzonder wanneer dependencies worden gepatched, als binary worden gedownload of vanaf de source code worden gebouwd (zie [7↗]). Controleer hierbij of de geproduceerde SBOMs compleet zijn en of ze geschikt zijn voor verder gebruik in vulnerability management processen. Inmiddels bestaan er al verschillende richtlijnen voor wat er in SBOMs dient te staan, zoals advies van de Amerikaanse overheid [13↗] en een technische richtlijn van het Duitse BSI [14↗].

4 Producing attestations | PyPI Docs
5 Generating provenance statements | npm Docs
6 Projecten als OpenSSF Scorecard houden deze informatie bij voor open-source software

Produceren

Fase 1.

Terug naar startfiguur

Extern geproduceerde SBOMs

Om de risico's op ontbrekende of incomplete SBOMs van producenten af te dekken kunnen organisaties nieuw SBOM-specifiek beleid invoeren binnen hun supply chain management. Een belangrijk fundament hiervan zou moeten zijn om duidelijke afspraken te maken met softwareleveranciers. Een goed beginsel hiervan zou zijn om altijd de beschikbaarheid van SBOMs mee te nemen als vereiste bij het aankopen van nieuwe software-producten.

Daarnaast is het van belang om het SBOM-ontvangstproces duidelijk af te stemmen; voornamelijk de methode waarop SBOMs verstuurd worden zodat malafide actoren een kleinere kans hebben om hun aanvallen te introduceren. Dit zou echter wel vereisen dat deze afspraken ook consistent nageleefd worden, omdat er anders geen verwachtingspatroon is waar een aanvaller buiten zou kunnen vallen. Hierop wordt in het hoofdstuk Delen verder ingegaan.

Door aangeleverde SBOMs te vergelijken met een zelfgemaakte versie zouden inconsistenties geïdentificeerd en gerectificeerd kunnen worden. Hiervoor kunnen Software Composition Analysis (SCA) tools gebruikt worden, die een overzicht kunnen geven van de open-source componenten van een software-product[7]. Deze tools worden doorgaans ook gebruikt om SBOMs te creëren. Hiermee wordt dit werk dus eigenlijk dubbel uitgevoerd, zowel door de leverancier als door de afnemer, met hetzelfde risico op gemiste dependencies. Daarnaast beschikt een afnemer vaak over minder informatie dan de leverancier. Bijvoorbeeld omdat ze geen beschikking heeft over de source code inclusief bestaande dependency informatie, maar alleen over de resulterende binary, waarvan de samenstelling moeilijk is te controleren. Een betere manier om voor transparantie en vertrouwen te zorgen, is wanneer de leverancier bepaalde garanties kan geven over de integriteit van het softwareproduct en de SBOM, door middel van de eerder genoemde attestaties.

7 Enhancing Software Security Through Software Composition Analysis (sbom.observer)

Produceren

Fase 1.

Terug naar startfiguur

Waar moet je rekening mee houden?

Bij het inrichten van de fase Produceren is van belang om te zorgen dat de SBOMs die worden geproduceerd een correct en compleet overzicht geven en alle benodigde informatie bevatten. Hiertoe kunnen de volgende stappen worden gezet:

- Controleer bij het generen en ontvangen van SBOMs voor *software artefacten* in uw beheer of de SBOM volledig is en zo niet welke informatie ontbreekt of onvolledig is. Controleer ook het format van de genereerde of ontvangen SBOM: SPDX, CycloneDX of een ander format.
- Controleer of de minimale elementen van de SBOMs zijn ingevuld en correct zijn: Supplier, Component Name, Component Version, Unique Identifier, Dependency Relationship, Author, Timestamp (zie [13↗]).
- Zorg bij de opslag van SBOMs voor passende autorisatie zodat alleen medewerkers of processen die gebruik maken van SBOMs voor verschillende taken binnen uw organisatie toegang hebben.
- Zorg bij de opslag van SBOMs voor passende cryptografische versleuteling om te voorkomen dat aanvallers informatie over het *software artefact* en mogelijke kwetsbaarheden bemachtigen.
- Het digitaal signen van een SBOM beschermt de authenticiteit en integriteit van een SBOM. Er zijn inmiddels verschillende tools die het signen van SBOMs mogelijk maken. Controleer of de SBOM een correcte digitale signature heeft.

Fase 2: Delen →

Delen

Fase 2.

Terug naar startfiguur

Delen

In veel gevallen zal, na het produceren, een SBOM tussen een producent en een afnemer worden gedeeld. Zekerheid bij de uitwisseling van deze SBOMs moet daarom ook goed geregeld zijn. Met de volgende functionele doelen moet daarom rekening worden gehouden:

Tabel 4: functionele doelen voor de deelfase.

Functioneel doel	Beschrijving
Sharing	Faciliteren van delen van een leverancier naar afnemer, of van een afnemer naar een derde.
Ingestion	Het importeren van SBOMs van leveranciers en/ of derden.

Het faciliteren van delen is op dit moment complex. Er zijn verschillende oplossingen beschikbaar met hun eigen voors en tegens. Een optie is het openbaar publiceren van de SBOM op het internet. Vervolgens kan deze daar permanent blijven bestaan.

Het nadeel hierbij is dat de SBOM dan voor eenieder inzichtelijk is. Ook voor een aanvaller of concurrent. Dit is iets waar veel leveranciers huiverig voor zijn. Zowel wanneer een SBOM publiekelijk ter beschikking wordt gesteld of alleen met specifieke partijen wordt gedeeld, moet vervolgens nog wel aan de vereiste securitydoelen worden voldaan.

Tabel 5: security doelen voor de deelfase.

Security doel	Beschrijving
Authenticity	Dit omvangt zowel het verifiëren van de identiteit van de afzender en de ontvanger, als het verifiëren van de juistheid van een SBOM.
Integrity	Een SBOM moet onaangepast bij de ontvanger terechtkomen.
Non-repudiation	De generatie of het delen van een SBOM wordt vastgelegd en kan niet achteraf worden ontkend door de leverancier.
Confidentiality	Alleen geautoriseerde partijen mogen een SBOM inzien.
Authorization	Alleen een geautoriseerde partij moet een SBOM kunnen delen en ontvangen.

Delen

Fase 2.

Terug naar startfiguur

Hoewel er in de deelfase een flink aantal security doelen van toepassing zijn, kunnen deze in theorie vervuld worden met bestaande tooling die gegevensuitwisseling mogelijk maakt. Veilige uitwisseling van gegevens is immers geen nieuw probleem. Vanwege het ontbreken van een standaardmanier voor het uitwisselen van SBOMs, vereist dit voorlopig inspanning van zowel de verzender als de ontvanger voor het opzetten van een maatwerkoplossing.

- **Authenticity** en **Integrity** kunnen worden gerealiseerd door een SBOM te ondertekenen. Hierdoor kan de ontvangende partij de identiteit van de ondertekenaar verifiëren en kan zij zich ervan verzekeren dat er tijdens het transport geen veranderingen zijn aangebracht in de SBOM.
- Voor **Non-repudiation** is het van belang dat een afnemer duidelijk weet dat het over de enige juiste (of meest recente) SBOM voor een bepaald artefact beschikt.
- **Confidentiality** en **Authorization** zou gegarandeerd kunnen worden door een autorisatie te vereisen alvorens de SBOM wordt verstuurd, bijvoorbeeld door middel van een API met authenticatie, en/of door de SBOM versleuteld op te sturen volgens een vooraf bepaalde methode.

Delen

Fase 2.

Terug naar startfiguur

Waar loop je tegenaan?

SBOMs worden soms gezien als een roadmap voor aanvallers, en dit is gedeeltelijk waar. Echter hebben potentiële aanvallers verschillende andere manieren om dergelijke informatie te achterhalen, zoals door te kijken naar bijgevoegde licenties van deelproducten of zelf uitgevoerde Software Composition Analysis. Het gevaar voor verdedigers van het ontbreken van SBOMs is veel groter dan het gevaar op misbruik [15↗]. De verdediger beheerd immers een groot assortiment aan software en moet snel kunnen ingrijpen bij nieuwe kwetsbaarheden, terwijl een aanvaller tijd heeft om een specifieke aanval voor te bereiden. Het maken van de juiste afspraken met leveranciers en bepalen onder welke voorwaarden zij bereid zijn om SBOMs te delen is dus zeer belangrijk. Afspraken over hoe de authenticiteit en integriteit van de SBOMs worden bewaakt mogen hierbij niet ontbreken.

Dit neemt echter niet weg dat SBOMs wel waarde kunnen hebben voor aanvallers. De methodes die de verdedigers gebruiken om kwetsbaarheden te linken aan hun SBOMs kunnen namelijk ook gebruikt worden door malafide actoren om hun aanvallen voor te bereiden. De SBOMs voor sensitievere software zullen waarschijnlijk niet publiek beschikbaar worden gemaakt, maar dit hoeft niet voor alle onderliggende onderdelen te gelden. Denk bijvoorbeeld aan open-source packages, waarvan een organisatie er recent de SBOMs van 72 miljoen softwareproducten in een database beschikbaar heeft gemaakt[8] of aan de informatie die beschikbaar is op bijvoorbeeld deps.dev en libraries.io. Aangezien de meeste organisaties gebruik maken van open-source software (98% uit een survey door Linux in 2021[16↗]), bestaat er een reëel risico dat een aanvaller deels zicht heeft op de samenstelling van de software die bij een organisatie in gebruik is.

Op dit moment worden veel SBOMs nog via email gedeeld [17↗], maar deze oplossing is lastig te automatiseren, te schalen en adequaat de beveiligen. Enkele experimentele tools maken het mogelijk om een API-instantie op te zetten waar geautoriseerd nieuwe SBOMs kunnen worden opvragen[9], maar deze zijn nog niet klaar voor inzet in operationele processen. Op dit moment zullen er dus maatwerk afspraken moeten worden gemaakt tussen afnemers en leveranciers.

8 SOOS Creates Public SBOM Database | SOOS
9 CycloneDX/transparency-exchange-api | A standard API specification for exchanging supply chain artifacts and intelligence

Hoe ga je hiermee om →

Delen

Fase 2.

Terug naar startfiguur

Hoe ga je hiermee om?

Het publiek beschikbaar zijn van informatie over de samenstelling van een softwareproduct is in veel gevallen een voldongen feit, of dit nu in het formaat van een SBOM is of handmatig door een aanvaller moet worden uitgezocht. Organisaties zullen moeten zorgen dat zij op zijn minst over dezelfde informatie beschikken als een mogelijke aanvaller zodat ze optimaal kunnen reageren al er een aanval plaats zou vinden. Leveranciers kunnen de afweging maken of zij hun SBOMs publiek beschikbaar stellen of dat ze SBOMs alleen met specifieke klanten delen. De consensus in de SBOM-community dat het tot beschikking hebben van SBOMs opweegt tegen het risico dat een aanvaller mogelijk toegang krijgt tot dezelfde informatie [15↗].

Ontbreken standaarden & best practices

Het grootste probleem waar organisaties op dit moment tegenaan zullen lopen is het gebrek aan vastgestelde best practices en het ontbreken van tooling voor het geautomatiseerd en op schaalbare manier delen van SBOMs. Idealiter zou het voor leveranciers makkelijk moeten zijn om SBOMs veilig aan te bieden aan afnemers, en voor afnemers makkelijk moeten zijn om SBOMs veilig af te nemen bij verschillende leveranciers. Zoals hierboven vermeld, wordt er op dit gebeid geëxperimenteerd. Ook wordt er vanuit verschillende overheden adviezen gepubliceerd [18↗]. Vooralsnog bestaan er nog geen volwassen of breed ondersteunde oplossingen, iets wat de adoptie van SBOM in de weg kan staan. De noodzaak om nu nog te vertrouwen op e-mails, secure transfer of maatwerkoplossingen voor het delen van SBOM en het ontbreken van best practices brengt ook risico's met zich mee. Organisaties zullen hiermee zelf moeten nadenken over en zorgen voor adequate beveiliging.

Waar moet je rekening mee houden? →

Delen

Fase 2.

Terug naar startfiguur

Waar moet je rekening mee houden?

Totdat er meer ontwikkelingen plaats hebben gevonden op het gebied van het delen van SBOMs, zullen organisaties zelf oplossingen moeten verzinnen en afspraken met hun suppliers of en afnemers moeten maken. Hiervoor moet er met de volgende zaken rekening worden gehouden:

- Het delen van SBOMs kan noodzakelijk zijn om afnemers te voorzien van informatie of inzichten te verschaffen binnen de organisatie over bijvoorbeeld kwetsbaarheden.
- Maak duidelijke afspraken over het proces en de methoden om SBOM met verschillende stakeholders te delen en zorg dat de afspraken ook zijn getest en werkbaar zijn.
- Zorg dat de methode om de SBOM te delen over voldoende beveiligingsmaatregelen beschikt, zoals 2FA.
- Ga uit van een *least privilege* principe waarbij enkel noodzakelijke informatie wordt gedeeld om het doel te bereiken. Deel bijvoorbeeld de SBOM van een bepaald softwareproduct alleen met de afnemers van dat product.
- Bij het delen van informatie over kwetsbare componenten kan de VEX standaard helpen om aan te geven wat de status van een kwetsbaar component is zonder de SBOM zelf te delen met stakeholders.

Fase 3: Beheren

Beheren

Fase 3.

Terug naar startfiguur

Beheren

De beheerfase beschrijft hoe een organisatie haar SBOMs bewaart en bijhoudt. Dit betekent dat naast het correct opslaan van SBOMs het ook van belang is dat organisaties binnen hun processen voor het updaten van hun softwareproducten, zoals security patches, bug-fixes en nieuwe functionaliteiten, ook een onderdeel opnemen om dergelijk veranderingen door te voeren in de bijbehorende SBOMs. In deze fase zijn de volgende functionaliteiten van belang:

Tabel 6: functionele doelen voor de beheerfase.

Functioneel doel	Beschrijving
Quality assessment & validation	Controleren van structurele juistheid en compleetheid van de SBOM in vergelijking met SBOM-standaarden en interne requirements.
Conversion	Informatie in SBOMs converteren naar formaten (vaak CycloneDX en SPDX) die worden ondersteund door de tooling en processen van de organisatie.
Management	Beheren van SBOM-collectie: zorgen voor beschikbaarheid, integriteit en confidentialiteit, en controleren van onder andere de geldigheid van SBOMs in bezit en het archiveren van SBOMs die niet langer relevant zijn voor de organisatie.
Enrichment	Verrijken van SBOMs met nieuwe kwetsbaarheidsinformatie en securityadviezen, bijvoorbeeld in de vorm van VEX- en CSAF-documenten, of het aanvullen van SBOM met dependency en kwaliteitsinformatie uit andere bronnen.

De eerste twee doelen zijn noodzakelijk, maar redelijk rechttoe rechtaan. Veel tooling kan werken met de meest gebruikte standaarden (CycloneDX en SPDX) en anders is het omzetten tussen deze standaarden in de meeste gevallen makkelijk te realiseren. Het daadwerkelijk 'beheren', de management functionaliteit, wordt met de huidige tooling in mindere mate vervuld.

Het meest populaire open-source dashboard om SBOMs in te managen is Dependency Track[10]. Deze richt zich voornamelijk op beheer- en verrijkingsfunctionaliteit, en slechts gedeeltelijk op een volledig veilige life cycle inclusief (automatische) kwaliteits- en integriteitschecks. Hiernaast zijn er nog enkele commerciële aanbieders actief.

10 Dependency-Track | Software Bill of Materials (SBOM) Analysis | OWASP

Beheren

Fase 3.

Terug naar startfiguur

De tooling in staat moeten zijn om verzamelde SBOM-informatie aan te vullen met nieuwe of externe veiligheidsinformatie. Bijvoorbeeld wanneer nieuwe CVEs bekend worden gemaakt, VEX-documenten door een leverancier worden gedeeld, of CSAF-advisories door een nationale CSIRT worden gepubliceerd[11]. De beheer-tooling zou de volgende security doelen moeten nastreven.

Tabel 7: security doelen voor de beheerfase.

Security doel	Beschrijving
Confidentiality	De opslag met SBOMs binnen een organisatie moet alleen toegankelijk zijn voor geautoriseerde gebruikers.
Integrity	Tijdens het beheer van een SBOM moet deze accuraat en compleet blijven. Hieronder valt ook dat updates in het softwareproduct doorgevoerd worden in de bijbehorende SBOM. Echter dient de organisatie wel de oude SBOM te archiveren, bijvoorbeeld om te identificeren hoe lang een net ontdekte kwetsbaarheid het softwareproduct heeft kunnen beïnvloeden.
Availability	De opslag met SBOMs moet altijd beschikbaar zijn om in te zetten.
Authorization	De SBOM moet alleen beheerd worden door een entiteit met de juiste autorisatie daarvoor.

Deze security doelen zullen voor de meeste organisaties bekend zijn en feitelijk weinig afwijken van de security doelen van andere interne tooling. Vanwege de beperkte beschikbaarheid van tooling is het echter lastig in te schatten in hoe gemakkelijk op dit moment ook aan deze security doelen voldaan kan worden.

- **Confidentiality** en **Authorization** is te bewerkstellingen door vast te stellen wie binnen de organisatie toegang dient te hebben tot de SBOMs. Juiste beveiliging maakt het voor aanvallers moeilijker om informatie over de IT-omgeving te verkrijgen en geeft het security personeel inzicht die zij nodig hebben voor het beschermen ervan. Goede access management en de juiste toekenning van rollen is hierbij van belang.

11 Voor uitleg over VEX en CSAF, lees het hoofdstuk "SBOM en VEX Integreren in de Vulnerability Management Cycle" in de startergids

Beheren

Fase 3.

Terug naar startfiguur

- **Integrity** van SBOMs *na inname in de tooling* lijkt in het huidige aanbod een bijzaak. Een sterke koppeling tussen de binnengekomen SBOM en de informatie in de beheertooling lijkt noodzakelijk om dit te bewerkstellingen. Daarnaast kunnen opnieuw attestaties van nut zijn. Deze helpen te bewijzen waar, wanneer, en hoe een artefact tot stand is gekomen. Hiermee kan dus worden gecontroleerd waar de SBOM-informatie vandaan is gekomen en of deze op een later moment is aangepast.

- **Availability** van de SBOMs zal belangrijker worden naarmate deze meer worden ingezet in security en compliance processen. Het inrichten van de beheertools dient zodanig te worden vormgegeven dat (offline) back-ups altijd beschikbaar zijn.

Beheren

Fase 3.

Terug naar startfiguur

Waar loop je tegenaan?

Zoals genoemd in de vorige fases is het van belang dat een organisatie beschikt over SBOMs die de juiste informatie bevatten en compleet zijn. Omdat SBOMs van verschillende producenten afkomstig zijn, kan de kwaliteit van de SBOMs wel eens fluctueren. Daarnaast kunnen SBOMs ook op verschillende manieren bij de organisatie binnen komen en in verschillende formats geleverd worden. Het is dus handig dat de kwaliteitscontrole voor zowel intern als extern geproduceerde SBOMs op dezelfde manier kan plaatsvinden. Hierdoor is het controleren van de kwaliteit van de SBOMs een belangrijke stap in de beheerfase.

Voor de effectieve inzet van SBOMs bij security en compliance processen is het van belang dat de informatie in de opgeslagen SBOMs ten aller tijde beschikbaar is en up-to-date worden gehouden; d.w.z. dat nieuwe SBOMs beschikbaar zijn op het moment dat software binnen de organisatie wordt geüpdatet of vervangen, dat SBOMs van niet langer gebruikte software na een bepaalde tijd wordt gearchiveerd. Hiernaast zullen de SBOMs direct moeten worden gekoppeld aan de nieuwste kwetsbaarheidsinformatie zodra deze bekend wordt. Als dit niet of onvoldoende gebeurt loopt de organisatie het risico op *false positives*, waardoor onnodige moeite wordt gestoken in het oplossen van irrelevante kwetsbaarheden, of *false negatives*, waardoor (belangrijke) kwetsbaarheden niet bekend en opgelost worden door de organisatie.

Dit probleem wordt nog eens verergerd door het feit dat het specifiek en uniek identificeren van software(componenten) een moeilijk probleem is. Er bestaan verschillende standaarden[12] en welke wordt gebruikt verschilt soms per toepassing. Het matchen van kwetsbaarhedeninformatie aan de informatie in SBOMs is hiermee geen uitgemaakt zaak (zie ook [19↗]).

Wanneer er geen sprake is van de centrale opslag van SBOMs, kan een situatie zich voordoen waarbij de toenemende hoeveelheid SBOMs binnen een organisatie op den duur verwarring veroorzaakt, bijvoorbeeld doordat de plek van opslag niet duidelijk geregistreerd wordt.

12 Zie ook het verdiepend kader “Uniek identificeren van software” in de startergids

Hoe ga je hiermee om →

Beheren

Fase 3.

Terug naar startfiguur

Hoe ga je hiermee om?

Om te garanderen dat een afnemer altijd over de juiste SBOMs beschikt, moeten hier met de leverancier duidelijke afspraken over worden gemaakt. Wanneer nieuwe versies van een softwareproduct beschikbaar wordt voor de afnemer, dient er per direct ook corresponderende SBOM voor deze versie beschikbaar te zijn. Een manier om dat te realiseren is om af te spreken dat SBOMs altijd direct worden meegeleverd met software updates, bijvoorbeeld doordat de SBOM in de software zelf kan worden opgevraagd, of doordat ze (geautomatiseerd) opgevraagd kunnen worden bij de leverancier [6↗].

Om nog meer waarde te halen uit de informatie in (ontvangen) SBOMs, zal deze moeten worden gekoppeld aan informatie uit andere bronnen. De meest voor de hand liggende zijn bronnen met kwetsbaarhedeninformatie. Het liefst in machine-readable vorm. Hier gaan we in de laatste fase verder op in. Voor meer inzicht in de kwaliteit van de software en het identificeren van potentiële risico's kunnen ook echter andere informatiebronnen worden aangesproken. Projecten als het eerdergenoemde OSSF Scorecard kunnen inzicht geven in de (bouw)kwaliteit van softwarecomponenten. Ook de leeftijd van dependencies en het beschikbaar zijn van updates kan een organisatie inzicht geven in de kwaliteit van de softwareproducten die zij binnenhalen. Een extra controle op de compleetheid van SBOMs kan worden gedaan door de dependencyinformatie in SBOMs te vergelijken met die in openbare bronnen als online repositories en deps.dev. Het verrijken van de informatie in SBOMs met andere bronnen wordt al door verschillende beheertools ondersteund.

Vanuit de SBOM-hoek is verder nog weinig bekend over best practises aangaande het beheerproces van SBOMs; d.w.z., zouden deze beter centraal of gedistribueerd opgeslagen kunnen worden? Hoe moeten back-ups beheerd worden? Hoelang dient een SBOM bewaard te blijven? Hoe ziet toegangsbeheer eruit voor SBOMs? Etc. Een concrete aanbeveling zou wel zijn om altijd een offline back-up voor systemen die ook zonder online integratie moeten kunnen functioneren te hebben. Daarnaast is het van belang dat de SBOM-beheeromgeving voorzien is van voldoende beveiligsmaatregelen, afhankelijk van de sensitiviteit van de SBOMs.

Waar moet je rekening mee houden? →

Beheren

Fase 3.

Terug naar startfiguur

Waar moet je rekening mee houden?

Bij het beheren en beschikbaar stellen van SBOMs voor gebruik in de eigen organisatie, dient er met de volgende zaken rekening te worden gehouden:

- De kwaliteit van SBOMs van verschillende software artefacten is een bekend probleem. Via de OWASP BOM Maturity Model kunnen gebruikers verschillende kwaliteitsonderdelen van SBOMs meten. Er bestaan verschillende tools om de kwaliteit van SBOMs te controleren.
- SBOMs worden regelmatig geüpdatet. Zorg voor een geautomatiseerd en getest updateproces voor SBOMs binnen de organisatie.
- Zorg ook tijdens het beheer van SBOMs voor passende autorisatie zodat alleen medewerkers die gebruik maken van SBOMs voor verschillende taken binnen uw organisatie toegang hebben.

Fase 4: Inzetten

Inzetten

Fase 4.

Terug naar startfiguur

Inzetten

Bij de daadwerkelijke inzet komen de voordelen van de SBOM het meest tot haar recht: in deze fase wordt duidelijk of en welke kwetsbaarheden van toepassing zijn voor de softwareproducten die bij een organisatie worden gebruikt. Daarom is het functionele doel als volgt:

Tabel 8: functionele doelen voor de inzetfase.

Functioneel doel	Beschrijving
Vulnerability assessment	Het inzetten van SBOM- en kwetsbaarheidsinformatie voor security operaties en risicobeheersing.

Hoewel SBOMs in principe ook in kunnen worden gezet voor bijvoorbeeld compliance management op gebied van licenties, of voor het uitpluizen van potentiële risico's in de gehele software supply chain, zullen veel organisaties de SBOMs primair willen inzetten voor vulnerability management processen. Daarom zullen we het in dit hoofdstuk alleen over dit functionele doel hebben.

Met de huidige tooling kunnen organisaties al SBOMs inzetten voor vulnerability management. Deze functionaliteit zit voornamelijk geïntegreerd in de tooling die ook voor SBOM-beheer wordt gebruikt. Veelal is de tooling in staat om SBOMs (automatisch) te koppelen aan bekende CVEs en te linken aan VEX- en CSAF-documenten. Hierbij zijn security doelen van belang die overeenkomen met die in de beheerfase.

Tabel 9: security doelen voor de inzetfase.

Security doel	Beschrijving
Confidentiality	De SBOM- en kwetsbaarheidsinformatie zou niet voor onbevoegden inzichtelijk moeten zijn.
Integrity	Het moet zeker zijn dat de ingezette SBOM correct en volledig is.
Availability	Een organisatie moet altijd beschikking hebben over de SBOM-collectie om daar kwetsbaarheden snel aan te kunnen linken.
Authorization	Alleen geautoriseerde personen mogen wijzigingen aanbrengen of informatie toevoegen aan de systemen.

Inzetten

Fase 4.

Terug naar startfiguur

Net als in de beheerfase zullen deze security doelen voor veel organisaties bekend zijn. Wederom richt de bestaande tooling zich voornamelijk op functionele doelen, en is het onduidelijk in hoeverre op dit moment de security doelen kunnen worden gerealiseerd.

- **Integrity** en **Availability** zullen met name moeten worden gewaarborgd door de juiste beveiligingsmaatregelen in de (back-end van de) beheertooling.

- **Confidentiality** en **Authorization** is net als in de vorige fase afhankelijk van de juiste toekenning van rollen en rechten.

Inzetten

Fase 4.

Terug naar startfiguur

Waar loop je tegenaan?

Net als in de vorige fase is het van belang dat informatie correct, up-to-date, en toegankelijk is. Daarnaast dient wederom te worden voorkomen dat ongeautoriseerde personen toegang hebben tot de SBOM- en kwetsbaarheidsinformatie of hierin wijzigingen aan kunnen brengen.

Om de adoptie van SBOMs aan te jagen, wordt er veelal gepromoot om meer te werken met SBOMs en aanvullende (machine-readable) dreigingsinformatie. Hoewel dit op zichzelf een goed idee, is het zaak om wel te voorkomen dat het gehele vulnerability management proces binnen een organisatie afhankelijk wordt van SBOMs en informatie afkomstig van leveranciers, zeker wanneer er geen duidelijkheid is over de correctheid van deze informatie. Hierdoor zou namelijk een situatie voor kunnen komen waarin een kwetsbaarheid wel bekend is, maar niet opgemerkt wordt door het security team van een getroffen organisatie, bijvoorbeeld doordat hun SBOMs incompleet zijn of doordat de manier van bekendmaken van de kwetsbaarheid niet past in het geautomatiseerde proces van de organisatie. Hierdoor zou de organisatie kwetsbaar kunnen blijven voor bekende kwetsbaarheden, die geëxploiteerd zouden kunnen worden door aanvallers.

Hoe ga je hiermee om →

Inzetten

Fase 4.

Terug naar startfiguur

Hoe ga je hiermee om?

Een volledige omschrijving van de in te richten operationele security processen valt buiten de scope van dit document. Het is vooral van belang dat SBOMs een aanvulling vormen op bestaande security processen om zo meer automatisering te kunnen bewerkstellingen en security personeel in staat te stellen om hun werkzaamheden efficiënter uit te voeren. Om SBOMs effectief in te kunnen zetten voor vulnerability management processen, is het van belang dat een SBOM snel kan worden opgevraagd en gekoppeld kan worden aan nieuwe dreigings-informatie. Dit vereist dat security personeel toegang heeft tot de SBOM-beheertooling, of dat er koppelingen worden gerealiseerd met bijvoorbeeld de Security Incident & Event Management (SIEM) tooling van de organisatie.

Waar moet je rekening mee houden? →

Inzetten

Fase 4.

Terug naar startfiguur

Waar moet je rekening mee houden?

Bij het inrichten van de inzet fase zouden de volgende aandachtspunten in overweging kunnen worden genomen:

- Zorg dat er een mogelijkheid is om SBOMs te kunnen matchen met ontvangen kwetsbaarhedeninformatie.
  - Het NCSC publiceert haar kwetsbaarhedenadvisories in CSAF-format
- Tooling voor het geautomatiseerd matchen van SBOMs met advisories die de CSAF standaard gebruiken is nog in ontwikkeling, blijf op de hoogte van de huidige stand van zaken.
- Maak gebruik van meerdere hoogkwalitatieve bronnen voor kwetsbaarhedeninformatie.
- Controleer geautomatiseerde processen op fouten en test ze regelmatig, bijvoorbeeld wanneer er langere tijd geen security advisories over binnenkomen.
- Richt in het proces back-up opties in voor het gebruik van security advisories die human-readible zijn.

Nawoord →

Terug naar startfiguur

Nawoord

Het is duidelijk dat het inrichten van een veilige en betrouwbare SBOM life cycle een complex en organisatie-overstijgend proces is. Hiervoor zullen afspraken gemaakt moeten worden tussen leveranciers en afnemers, zal er tooling moeten worden aangeschaft of ontwikkeld, zullen processen moeten worden ingericht en personeel getraind. Al met al zal het voor veel organisaties dus een behoorlijke investering vergen. SBOMs en geautomatiseerde uitwisselingen van kwetsbaarheidsinformatie brengen echter veel voordelen, zoals een snellere en efficiëntere vulnerability management. Hierbij zullen SBOMs niet een silver bullet zijn, maar een belangrijke aanvulling op een volwassen vulnerability management proces.

In deze gids zijn verschillende functionele doelen en security doelen geïdentificeerd voor de verschillende fases in de life cycle. Op dit moment voorziet het aanbod van SBOM-gerelateerde tooling nog niet aan al deze doelen, of kan het veel inspanning vergen van organisaties om deze doelen met beschikbare tooling te realiseren. Daarentegen is er op dit moment wel sprake van een duidelijke (wild)groei in het aanbod. Er bestaan veel tools met dezelfde functionaliteiten, soms is dit zelfs het geval bij meerdere tools van een enkele aanbieder. Omdat er ook weinig zicht is op de kwaliteit van de tools, kan het voor organisaties lastig zijn om een te kiezen.

Voor een deel wordt het gebrek aan tooling verklaard doordat SBOMs nog relatief nieuw zijn, en nog niet op grote schaal zijn omarmd door leveranciers en afnemers. Een organisatie die op dit moment SBOMs wil verzamelen zal al snel tegen een leverancier aanlopen die niet in staat is om deze te leveren, of die vanwege vermeende veiligheidsrisico's of haar concurrentiepositie dit niet zal willen doen. Nieuwe wetgeving zoals de CRA [3] zal echter een stimulerend effect hebben op SBOM-adoptie. Wanneer deze adoptie toeneemt, zal dit op termijn ook leiden tot het vormen van best practices en plekken of communities waar deze gedeeld kunnen worden.

Naarmate SBOMs verder inburgeren en het tooling-aanbod steeds volwassener wordt, zal het voor organisaties steeds makkelijker worden om ook de SBOM te omarmen. Dat betekent niet dat organisaties op dit moment zich een afwachten houding kunnen aanmeten. Organisaties zouden zich juist nu al kunnen voorbereiden door kennis op te doen van SBOM, VEX, CSAF en andere gerelateerde standaarden, kijken of ze zich aan kunnen sluiten bij communities die SBOM-gebruik stimuleren, in gesprek gaan met leveranciers, of experimenteren met het genereren en inzetten van SBOMs voor intern ontwikkelde softwareproducten.

Bronnen

Bronnen

Terug naar startfiguur

[1] ENISA, „Foresight Cybersecurity Threats For 2030 - Update 2024: Extended report," ENISA, 2024.

[2] TNO en NCSC, SBOM Startersgids, NCSC, 2023.

[3] European Commision, „Cyber Resilience Act," 20 11 2024. [Online]. [Geopend 1 5 2025].

[4] L. Kohnfelder en P. Garg, The threats to our products, Microsoft, 1999.

[5] B. van Riel, S. Kuijpers en R. de Koning, „Using the Software Bill of Materials for Enhancing Cybersecurity," Capgemini Invent, 2021.

[6] CISA, „Software Bill of Materials (SBOM) Sharing Lifecycle Report," CISA, 2023.

[7] S. Yu, W. Song, X. Hu en H. Yin, „ On the correctness of metadata-based SBOM generation: A differential analysis approach," in 54th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN), 2024.

[8] B. Kloeg, A. Ding, S. Pellegrom en Y. Zhauniarovich, „Charting the path to SBOM adoption: A business stakeholder-centric approach," in Proceedings of the 19th ACM Asia Conference on Computer and Communications Security, 2024.

[9] The Linux Foundation, „Supply chain threats," [Online].

[10] OWASP, „OWASP Software Component Verification Standard," [Online].

[11] in-toto Authors, „in-toto specification," [Online].

[12] The Linux Foundation, „Sigstore," [Online].

[13] NTIA, „The Minimum Elements For a Software Bill of Materials," The United States Department of Commerce, 12 7 2021. [Online].

[14] BSI, „Technical Guideline TR-03183: Cyber Resilience Requirements for Manufacturers and Products Part 2: Software Bill of Materials (SBOM)," BSI, 2024.

[15] NTIA Multistakeholder Process on Software Component Transparency, „SBOM Myths vs. Facts," [Online].

[16] S. Hendrick, „Software Bill of Materials (SBOM) and Cybersecurity Readiness," 01 2022. [Online].

[17] Cybersecurity & Infrastructure Security Agency, „Software Bill of Materials (SBOM) Sharing Lifecycle Report," CISA, 2023.

[18] CISA, „SBOM Sharing Roles and Considerations," 28 3 2024. [Online].

[19] The SBOM Forum, „A Proposal to Operationalize Component Identification for Vulnerability Management," 13 09 2022. [Online].

Terug naar startfiguur

Colofon

Deze gids is opgesteld binnen de meerjarige onderzoekssamenwerking van het NCSC en TNO naar het versterken van supply chain management.

Auteurs:

Niels Brink (TNO)
Hidde-Jan Jongsma (TNO)
Thomas Sierink (TNO)
Rik van Dijk (NCSC)
Dion Koeze (NCSC)
Bart Snijdelaar (NCSC)

Nationaal Cyber Security Centrum
Ministerie van Justitie en Veiligheid

TNO innovation for life