# Cyberveiligheid

## Een praktische gids

PEOPLEWARE
we make your software

## Waarom plots deze aandacht voor cyberveiligheid?

Cyberveiligheid is al een hele tijd een aandachtspunt, maar recent komt het onderwerp steeds frequenter aan bod.

Een belangrijke reden hiervoor is de sterke toename van dreigingen en effectieve incidenten in de afgelopen jaren. Deze groei wordt gedreven door onze toenemende afhankelijkheid van cybersystemen. Hierdoor neemt de kans op cyberincidenten door menselijke fouten toe, terwijl de impact ervan groter is dan ooit. Aanvalssoftware is toegankelijker, goedkoper, en eenvoudiger in gebruik. Malafide acties leveren gemiddeld meer financiële winst op dan voorheen, terwijl de pakkans nauwelijks is toegenomen.

Tegelijkertijd hebben geopolitieke ontwikkelingen geleid tot een stijging van gerichte aanvallen uitgevoerd door goed gefinancierde organisaties van allerlei strekking (in 2013 werd duidelijk dat de telefooncentrales van Belgacom gehacked waren door … de Britse geheime dienst).

Er is de laatste maanden niets fundamenteels veranderd omtrent cyberveiligheid. Cybersystemen hadden altijd al veiligheidsvereisten, sinds de doorbraak van het internet, en zelfs al daarvoor (virussen op diskettes), en de trends zijn al jaren duidelijk.

PeopleWare neemt dit al jaren ter harte door de best–practices van het moment toe te passen bij softwareontwikkeling. De wereld evolueert echter, technisch, en wettelijk.

De EU vaardigde in 2016 de NIS(1) en GDPR richtlijnen uit, die ons oplegden om cybersystemen veilig te maken, en de privacy van natuurlijke personen in de EU te verzekeren. In 2022 werd NIS(1) vervangen door de NIS2–richtlijn, die wordt gerealiseerd in Belgische wetgeving door de Wet tot vaststelling van een kader voor de cyberbeveiliging van netwerk- en informatiesystemen van algemeen belang voor de openbare veiligheid, die in werking trad op 18 oktober 2024 . In grote lijnen zegt de NIS2–richtlijn tegenover NIS(1) “we menen het”. Het maakt duidelijk wie verantwoordelijk is, en maakt de vraag die gesteld wordt concreter: “*stel mij gerust dat je over de veiligheid van je cybersystemen gedetailleerd hebt nagedacht, en dat jouw cybersystemen voor mij veilig zijn*”. In januari 2025 worden de zaken strenger voor

financiële instellingen onder DORA. De Cyber Resilience Act (CRA), van toepassing op hardwareproducten met digitale elementen die in de EU verhandeld worden en software die los wordt verkocht, werd gepubliceerd op 20 november 2024. Ze legt een aantal meldingsvoorwaarden op aan de producenten vanaf 20 februari 2026, en strikte vereisten vanaf 26 november 2027.

## Sterke groei van cyberdreigingen en –incidenten: objectieve cijfers

- 2023: een jaar van ongekende uitdagingen voor cybersecurity in België (Jaarverslag 2023 — Centre for Cybersecurity Belgium)
- CS-barometer — Maturiteit in cybersecurity bij Vlaamse bedrijven — situatie 2023 (ecoom – expertisecentrum voor O&O monitoring; Departement Economie, Wetenschap & Innovatie))
- Een derde meer cyberaanvallen bij Belgische organisaties (DataNews 2024-08-02)
- Recordpiek cyberaanvallen: Belgische organisaties incasseren 69 procent meer incidenten (DataNews 2024-10-21)

## Is deze regelgeving op mij van toepassing?

De Wet tot vaststelling van een kader voor de cyberbeveiliging van netwerk- en informatiesystemen van algemeen belang voor de openbare veiligheid (de Belgische toepassing van de NIS2–richtlijn), DORA, en de CRA maken duidelijk op welke organisaties ze rechtstreeks van toepassing zijn. Of je organisatie rechtstreeks onder de richtlijnen valt hangt af van de sectoren waarin je organisatie actief is, het soort diensten dat je organisatie levert, waar je gevestigd bent en waar je de diensten levert, en de grootte van je organisatie. We verwijzen graag naar de website van het Centre for Cybersecurity Belgium voor meer details:

- De NIS2-wet

- Waar u ook een makkelijke "scope assesment tool" kan vinden: De Cyber Resilience Act (CRA)

In geval van twijfel raden we aan uw raadsman te consulteren.

Maar ook als uw organisatie niet rechtstreeks onder de richtlijnen valt, zal u waarschijnlijk toch met deze richtlijnen geconfronteerd worden.

Enerzijds is cyberveiligheid belangrijk voor uw organisatie, onafhankelijk van wetten. De NIS2–richtlijn verwijst naar de ISO 27000–familie van standaarden die een excellent denkkader bieden om hier gestructureerd mee om te gaan, en gefundeerd gerust te zijn in de cyberveiligheid van uw systemen.

Verder is de GDPR-richtlijn sowieso van toepassing wanneer u enige Personal Identifiable Information (PII) verwerkt, en kan u enkel gerust zijn omtrent privacybehoud in systemen waarvan u a priori gerust bent dat ze cyberveilig zijn.

Het meest belangrijke is echter dat de NIS2-richtlijn, DORA, en de CRA ons terdege bewust maken van onze plaats in het economisch weefsel van de Europese Unie. Als de richtlijnen niet rechtstreeks op u van toepassing zijn, dan misschien wel op één van uw klanten, of klanten van uw klanten. De organisaties die uiteindelijk wel rechtstreeks onder de NIS2–richtlijn, DORA, of de CRA vallen worden onder deze richtlijnen verantwoordelijk gesteld voor hun ganse toeleveringsketen.

Wanneer een bestuurder van een organisatie die wel onder de richtlijnen valt moet aantonen dat ze gedetailleerd hebben nagedacht over de veiligheid van hun cybersystemen, zullen ze uw organisatie vragen om aan te tonen dat u gedetailleerd hebt nagedacht over de veiligheid van uw cybersystemen, zodat ze gerust kunnen zijn dat het geheel veilig is en de privacy bewaard wordt. Als u cybersystemen of –diensten levert is dit duidelijk. Maar als u van kritiek belang bent voor uw klant is uw algemene

cyber–weerbaarheid voor hen van belang. Zij moeten kunnen aantonen dat hun werking niet in het gedrang kan komen door problemen die zich potentieel bij u voordoen. En met uw antwoord op hun vragen komen een aantal verantwoordelijkheden en aansprakelijkheden. Dat is hoe de EU duidelijk maakt “we menen het”.

Bent u, of is één van uw directe of indirecte klanten een overheidsinstelling? Of groot genoeg, en doen uw klanten iets financieels, leveren ze IT-infrastructuur, zijn ze bezig met elektriciteit, olie, gas, waterstof, voedingsproductie of –verdeling, gezondheid, de productie van medische, computer, of elektrische apparaten, machines, auto’s of andere vervoermiddelen? Verzorgen ze vervoer, of zijn ze betrokken bij drink– of afvalwater, of afvalverwerking in het algemeen? Dan is NIS2 op hen van toepassing, en zal men, direct of indirect, ook van u vragen om gefundeerde geruststelling. Produceert één van uw directe of indirecte klanten een hardwareproduct met digitale elementen of software die los wordt verkocht? Dan is de CRA op hen van toepassing.

PeopleWare bevindt zich in deze positie. U bent één van onze klanten. U zal ons vragen aan een aantal dingen te voldoen.

## Wat is cyberveiligheid?

Sinds de jaren '80 van de vorige eeuw wordt cyberveiligheid beschreven als het bewaren van CIA, i.e.:

- vertrouwelijkheid (**C**onfidentiality),
- integriteit (**I**ntegrity), en
- beschikbaarheid (**A**vailability).

Vertrouwelijkheid zorgt ervoor dat gevoelige informatie alleen toegankelijk is voor degenen die bevoegd zijn om deze te bekijken. Dit principe is essentieel voor het beschermen van gegevens tegen ongeoorloofde toegang en openbaarmaking.

Integriteit richt zich op het behouden van de nauwkeurigheid en volledigheid van gegevens gedurende hun gehele levenscyclus. Dit principe zorgt ervoor dat informatie ongewijzigd blijft, behalve door bevoegde personen en processen.

Beschikbaarheid zorgt ervoor dat informatie en middelen toegankelijk zijn voor geautoriseerde gebruikers wanneer dat nodig is.

In de 21ste eeuw zijn we ons bewust geworden van het belang van het beschermen van privacy. We mogen geen persoonsgegevens (informatie die direct of indirect naar een natuurlijke persoon herleidbaar is; Personal Identifying Information — PII) verzamelen die niet strikt noodzakelijk zijn, en we mogen deze niet langer bewaren dan nodig. Persoonsgegevens mogen alleen worden gebruikt voor het doel waarvoor ze zijn verzameld en moeten adequaat beschermd worden om vertrouwelijkheid en integriteit te waarborgen.

Het is duidelijk dat privacy enkel beschermd kan worden in cybersystemen die a priori vertrouwelijkheid en integriteit bewaren.

Daarom vatten we cyberveiligheid samen als CIA&PP: Confidentiality, Integrity, Availability, and Privacy Preservation.

## Wat is de vraag?

De essentiële vragen die een organisatie die verantwoordelijk is voor cybersystemen dient te beantwoorden zijn:

- Ben ik gerust dat de cybersystemen onder onze verantwoordelijkheid vertrouwelijkheid, integriteit, beschikbaarheid, en privacy bewaren?
- Hoe kan ik mijn rust verantwoorden tegenover mijn klanten en gebruikers?

Het goede werk van de EU en de ISO 27000–familie van standaarden biedt een excellent denkkader om op deze vragen te antwoorden.

Het idee dat een cybersysteem veilig is zolang er geen kwetsbaarheid wordt aangetoond, is een misvatting. Het opsporen van potentiële problemen in duizenden lijnen code en infrastructuurconfiguratie is buitengewoon arbeidsintensief en afhankelijk van de creativiteit van de onderzoekers. Hoewel bestaande problemen soms met geluk, toeval, of een enorme inspanning, worden ontdekt, bewijst het niet–vinden van een kwetsbaarheid nooit dat er geen kwetsbaarheden bestaan. Hetzelfde geldt voor penetratietesten (PEN–testen). Eén van de doelstellingen van de Europese Unie is om deze kromredenering de wereld uit te helpen.

De vraag is niet of de cybersystemen onder uw verantwoordelijkheid absoluut veilig zijn — dat is onmogelijk om aan te tonen. Zelfs als dit vandaag zou lukken, gegeven de huidige bekende aanvalsmethoden en operationele risico's, dan zou dezelfde oefening morgen opnieuw uitgevoerd moeten worden, omdat het systeem evolueert en kwaadwillenden voortdurend nieuwe scenario’s ontwikkelen. Bovendien zijn de menselijke mogelijkheden om fouten te maken onbegrensd. Men spreekt in deze context van known knowns, known unknowns, en unknown unknowns.

Noteer ook dat de ISO 27000–familie van standaarden denkkaders zijn, en niet een lijst van regels die afgevinkt moeten worden. De nadruk licht op het denken, en de standaarden bieden een excellent kader waarin dat kan gebeuren. Het kader helpt om onszelf er van te overtuigen dat we volledig zijn.

Het is de taak van de organisatie om aan te tonen aan zichzelf, klanten, en gebruikers:

- dat voldoende is nagedacht over de risico's en mogelijke kwetsbaarheden met betrekking tot CIA&PP die haar cybersystemen met zich meebrengen, i.e., dat de known knowns, en known unknowns in kaart zijn gebracht en worden behandeld, en
- dat voldoende maatregelen zijn getroffen om die te mitigeren, en
- dat de organisatie zich bewust is van het bestaan van unknown unknowns.

## Verantwoord vertrouwen

Kort samengevat is het antwoord:

- het bepalen van het standpunt omtrent CIA&PP van uw organisatie,
- het in kaart brengen van alle componenten van de cybersystemen onder de verantwoordelijkheid van uw organisatie,
- het gestructureerd in vraag stellen van de CIA&PP–kwaliteiten van elk van de componenten en hun interacties, en
- het continu bewaken, verifiëren, en bijstellen van maatregelen die voor de CIA&PP van de componenten zorgen.

Documentatie van dit werk verantwoordt uw vertrouwen in de CIA&PP–kwaliteiten van de cybersystemen onder uw verantwoordelijkheid voor uzelf, en tegenover anderen.

Het uiteindelijk resultaat is een consistent, dekkende governance: een geheel van principes, richtlijnen (policies), sturingskader en beleidsmaatregelen, concrete procedures en vereisten (procedures en requirements), en uiteindelijk concrete beheersmaatregelen (controls), vanuit de Raad van Bestuur tot in de code, waarvan de naleving gecontroleerd wordt of verzekerd is. Wanneer de organisatie onderhevig is aan wetgeving of contractuele verplichtingen omtrent CIA&PP moet verder afgetoetst worden of dit geheel voldoet aan de gestelde vereisten (compliance), eventueel door een derde partij (certificatie).

## Een pragmatische aanpak

Wij zijn van mening dat een top-down aanpak niet het meest effectieve of efficiënte pad is om dit doel te bereiken. Vaak wordt een generiek kant-en-klaar beleid ingevoerd, bestaande uit een verzameling strenge, abstracte policies, procedures en requirements die zich vooral richten op triviale aspecten van de cyberinfrastructuur. Er is geen aandacht voor pragmatiek, concrete toepasbaarheid, of de toepasselijkheid op de unieke omstandigheden van de organisatie en haar unieke cybersystemen. Zelfs wanneer er geen certificeringsvereisten zijn, ligt de nadruk vaak op het behalen van compliance met minimale impact, in plaats van op een goed onderbouwde bescherming van de CIA&PP van de cybersystemen. Dergelijk beleid lijkt vooral gericht op het geruststellen van – al dan niet imaginaire – auditeurs, in plaats van op het daadwerkelijk beschermen van de organisatie tegen de risico's van cyberincidenten.

Volgens ons is het effectiever of efficiënter is om een bottom–up aanpak te volgen. Zorg ervoor dat de cybersystemen CIA&PP zijn:

- in elke versie
- door goede en gedocumenteerde keuzes en ontwerp,
- concrete technische beheersmaatregelen en
- automatische verificatie,
- gebaseerd op bestaande
    - inzichten van de brede cyberveiligheidsgemeenschap,
    - best–practices,
    - standaarden, en
    - instrumenten die nu beschikbaar zijn

en formuleer enkel meer–abstracte principes en richtlijnen die wat realiseerbaar is en gerealiseerd is. De policies, procedures, en requirements die je aanneemt voor de organisatie

zullen immers ook in de toekomst gelden. Ook legaal advies omtrent dataverwerking en – retentie is veel specifieker, toepasselijk en toepasbaar in de context van concrete applicaties.

Als de leden van de Raad van Bestuur van uw organisatie op deze basis gerust zijn dat de cybersystemen onder hun verantwoordelijkheid vertrouwelijkheid, integriteit, beschikbaarheid, en privacy bewaren, is dat verantwoord, en is het doel bereikt zonder onnodige inspanningen en frustratie. Indien nodig kan het geheel nog afgetoetst worden aan wetgeving of contractuele verplichtingen.

Met deze aanpak stellen we pragmatische en concrete Secure Application Development (SecAppDev) als prioritair boven vage doelstellingen waarvan het onduidelijk is hoe ze bereikt moeten worden.

## Enkel voor cybersystemen waarvoor een relevant risico bestaat

Voor elk cybersysteem onder de verantwoordelijkheid van de organisatie wordt bepaald wat het risico is voor de organisatie als zich een incident zou voordoen.

Enkel cybersystemen met een risico dat door de organisatie als significant wordt beschouwd, verdienen aandacht.

Het risico wordt bepaald door in te schatten wat de waarschijnlijkheid is dat een cyberincident zich voordoet (bijvoorbeeld: zeer onwaarschijnlijk, onwaarschijnlijk, mogelijk, waarschijnlijk, bijna zeker) en de impact die een dergelijk incident zou hebben op de organisatie, zowel rechtstreeks als onrechtstreeks (bijvoorbeeld: verwaarloosbaar, laag, gemiddeld, hoog, of extreem). De impact kan financieel zijn, legaal, of bijvoorbeeld reputatieschade omvatten. Het product van de waarschijnlijkheid en de impact vormt het risico.

Een lage waarschijnlijkheid gecombineerd met een kleine impact resulteert in een laag of verwaarloosbaar risico, terwijl een hoge waarschijnlijkheid met een grote impact leidt tot een hoog of extreem risico.

Voor cybersystemen waarvan de organisatie oordeelt dat ze aandacht verdienen kan in meer detail gegaan worden. Er bestaan lijsten van scenario's die de aandacht vestigen op soms onverwachte aspecten, zodat voor elk scenario afzonderlijk het risico kan worden ingeschat.

In het algemeen is het de organisatie die zelfstandig deze inschattingen maakt, afhankelijk voor haar risicobereidheid. Enkel wanneer de organisatie onder de NIS2–richtlijn, DORA, of de CRA valt, grijpt de EU in en stelt de regelgeving minimumvereisten en grenzen.

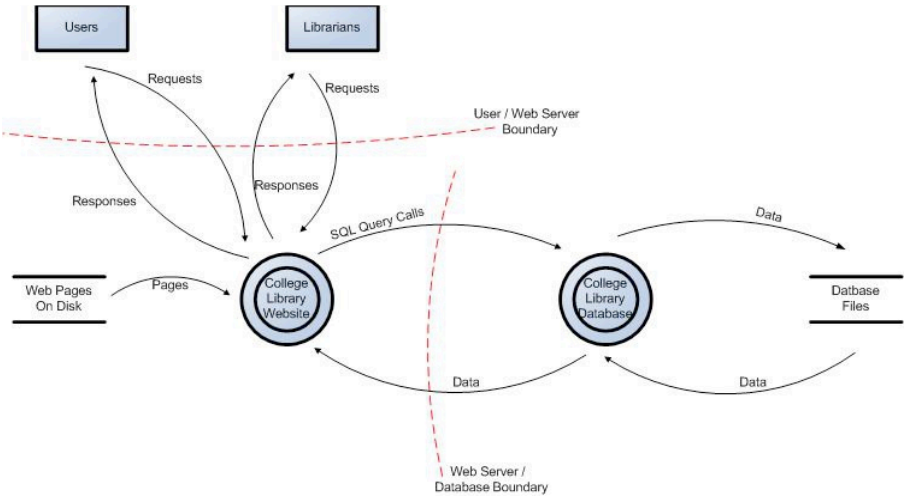
Wanneer u de resultaten van deze risico–inschatting documenteert, de oefening regelmatig herhaalt, en beschrijft hoe u omgaat met gedetecteerde risico's die groter zijn dan wat uw organisatie wil aanvaarden, heeft u al een belangrijk deel van het kader ingevuld dat de ISO 27000 familie van standaarden voorstelt (risk management policy).

## Maak kritische cybersystemen aantoonbaar vrij van bekende CIA&PP kwetsbaarheden

De enige bekende aanpak om onszelf verantwoord te overtuigen van het feit dat cybersystemen CIA&PP zijn, is een gestructureerde analytische en reductionistische aanpak. We brengen de mogelijke interacties met het cybersysteem, component per component, in kaart. Dit zijn de known knowns en de known unknowns. Voor elk van de mogelijke interacties met elk van de componenten verzekeren we er ons van dat misbruik niet mogelijk is, en een ongelukkige handeling niet tot een veiligheidsbreuk kan leiden.

### Threat model

Voor nieuwe ontwikkeling gebeurt dit best door de veiligheids– en privacy–bekommernissen mee te nemen van in de analyse en het ontwerp (secure–by–design, shift left). Dit pas na de feiten doen voor bestaande applicaties is mogelijk, maar veel minder efficiënt. Wanneer dit vanaf de eerste gesprekken meegenomen wordt, leidt dit meestal tot een hoge standaardisatie van de interactie–mogelijkheden, waarvoor de bekommernissen en mitigaties in een dreigingsmodel (threat model) één maal worden uitgewerkt per interactiepatroon. Wanneer dit niet vanaf het ontwerp werd meegenomen moeten alle interacties afzonderlijk bekeken worden. Dit is nog steeds mogelijk, maar vereist meer werk.




Threat models maken mogelijke CIA–dreigingen expliciet gebaseerd op STRIDE, en mogelijke privacy–kwesties gebaseerd op LINDDUN. Wanneer de kwesties expliciet zijn moeten soms keuzes gemaakt worden: in sommige gevallen conflicteren veiligheidsverwachtingen met privacy bekommernissen. Een goed voorbeeld is logging van bepaalde interacties. Voor CIA–redenen is het zinnig om bij te houden wie welke informatie opgevraagd heeft (repudiation). Voor privacy–redenen is het

misschien net nodig dat niet aangetoond kan worden dat iemand bepaalde informatie geraadpleegd heeft (non-repudiation).

Het thread model zorgt ervoor dat intrinsieke kwetsbaarheden expliciet gemaakt worden, en pas van zodra deze gekend zijn kunnen we ons ertegen verdedigen.

Waar schendingen van CIA&PP mogelijk zijn, wordt het ontwerp aangepast om ze te vermijden, of worden concrete technische beheersmaatregelen voorzien om de mogelijke schendingen te verhinderen. Er wordt een manier gespecifieerd om automatisch, herhaalbaar en bij elke nieuwe versie, te verifiëren dat aan de gestelde vereisten voldaan is. Zonder verificatie blijven de bepalingen dode letter. Maatregelen die niet eenvoudig herhaalbaar gecontroleerd kunnen worden hebben geen zin.

## Diepteverdediging

Het threat model brengt de known knowns en de known unknowns in kaart, specifieert maatregelen tegen mogelijke schendingen die in kaart gebracht zijn, en voorziet dat de verdedigingen tegen de bekende dreigingen inderdaad gerealiseerd zijn. Wat overblijft zijn de unknown unknowns. Voorbeelden zijn aanvalstrategieën of ongelukkige handelingen waar we niet aan gedacht hebben, aanvalstrategieën die later ontwikkeld worden en kwetsbaarheden in componenten die later ontdekt worden, configuratiewijzigingen met onbedoelde gevolgen, enzoverder.

Diepteverdediging (defense–in–depth) wordt gebruikt om hier zo veel mogelijk tegen te verdedigen. Alle beveiligingsmaatregelen, die componenten die we gebruiken voor de realisatie van de gevraagde functionaliteit, worden gebruikt. Elke interactie, functie, procedure, methode wordt ontworpen en gerealiseerd met aandacht voor algemene veiligheidsaspecten in isolatie. Er wordt niet gerekend op het feit dat de component of code–eenheid wel op een veilige manier gebruikt zal worden. De algemene veiligheidsaspecten zijn best practices die de gemeenschap verzamelt en beschikbaar stelt. Een erg belangrijk maar simpel voorbeeld van zo een best practice is dat eenvoudig leesbare code veiliger is dan moeilijk leesbare code.

Complexiteit is de vijand van beveiliging. Andere voorbeelden zijn dat het geen goed idee is om een paswoord letterlijk in de code op te nemen, input van een gebruiker te vertrouwen

zonder validatie, tekst van een externe bron te gebruiken als uitvoerbare code, of een manier van werken te gebruiken die het mogelijk maakt dat dat zou kunnen gebeuren (injection). Wanneer het gebruik van je code nu de zwakheid niet openstelt, is de kans reëel dat een andere ontwikkelaar, binnen 5 jaar, terwijl deze werkt aan een verdere evolutie van het systeem, zich niet bewust is van de zwakheid en je code wel op die manier gebruikt.

Een belangrijke bron van de best practices zijn de OWASP Cheat Sheets. Verder heeft de gemeenschap over de jaren in verschillende technische contexten heel wat anti–patronen herkend. Dit zijn code structuren die minder gepast of veilig zijn dan alternatieven. Voor moderne platformen bestaat er, vaak als Open Source beschikbaar, gereedschap dat ons wijst op bekende anti–patronen in de code die we schrijven, zodat we de code kunnen aanpassen (SAST). Dit is een markt in sterke ontwikkeling, waar meer geavanceerde oplossingen beschikbaar zijn. Deze diensten zijn relatief duur, en nemen het zekere voor het onzekere. Ze produceren redelijk wat vals– positieve waarschuwingen, die allemaal geverifieerd moeten worden. Anders heeft het gebruik van de dienst geen zin. Op dit moment van de evolutie van deze tools raden we aan om wat als Open Source beschikbaar is, of wat beschikbaar is in gereedschap dat sowieso tijdens ontwikkeling gebruikt wordt, tijdens de Secure Software Development Life Cycle (SSDLC) te gebruiken, en te overwegen de extra commerciële diensten in de SSDLC op te nemen voor cybersystemen met een hoger veiligheids– of privacy risico.

Diepteverdediging is een van origine militair concept, dat in vele sectoren wordt toegepast om veiligheid te verhogen. De nood voor defense–in–depth om systemen robuust en veilig te maken wordt goed gekaderd in How Complex Systems Fail (Richard I. Cook, 1998).

## Continue updates van gebruikte componenten

De toepassingen die PeopleWare maakt steunen op een heleboel externe middelen. De code die wij schrijven:

- is gericht op een specifiek platform (.net, Java, Angular, Node, …)
- maakt gebruik van externe en interne bibliotheken en frameworks

- wordt uitgerold op een specifieke infrastructuur en stack (een OS, een webserver, …, of een combinatie van cloud services) en
- wordt ontwikkeld met gereedschap dat deze platformen, bibliotheken, frameworks, infrastructuur en stack ondersteunt.

Elk van deze componenten evolueert, van versie naar versie.

### Versnelde, continue evolutie

De versies van componenten worden meestal uitgedrukt met een `major.minor.patch` schema, bijvoorbeeld `v3.6.9`.

- Nieuwe versies, bijvoorbeeld `v3.6.8`, hebben de intentie om niets aan de functionaliteit te veranderen, maar om functionele of veiligheidskwesties in de vorige versie op te lossen. Incompatibiliteiten zijn in principe niet toegestaan. Het is dus belangrijk om deze zo snel mogelijk toe te passen.
- Nieuwe versies, bijvoorbeeld `v3.7.0`, hebben de intentie om alle bestaande functionaliteit te behouden, maar voegen nieuwe mogelijkheden toe. Incompatibiliteiten zijn in principe niet toegestaan. De code die van de vorige versie afhangt gebruikt de nieuwe mogelijkheden logischerwijze niet, en dus is een update probleemloos wanneer alles als bedoeld gebeurt.
- Nieuwe versies, bijvoorbeeld `v4.0.0`, mogen bestaande functionaliteit verwijderen of veranderen.

Wanneer dit schema (semantic versioning) niet letterlijk gebruikt wordt, kan de evolutie tot deze principes herleid worden.

De versie–evolutie van deze componenten is de laatste jaren erg versneld. Waar er vroeger major updates waren om de 1, 2 of 3 jaar, met enkele minor en patch updates tussendoor om kleine gebreken op te lossen, is de wereld geëvolueerd naar quasi continue evolutie. Voor sommige externe bibliotheken is een wekelijkse update normaal. Major updates verlopen niet gepland voor niet–fundamentele componenten, maar gebeuren wanneer het werk klaar is. De platformen, operating systems, en andere componenten in de stack worden quasi continu

geüpdatet met minor en patch evoluties. Major updates worden vaak wel gepland, maar gebeuren ook veel frequenter dan vroeger. Cloud services evolueren vaak dagelijks of wekelijks. Er zijn major nieuwe versies die de platformen volgen. Het voordeel van cloud services is dat de kleine evoluties voor ons gedaan worden. Overstappen naar een nieuwe major versie moeten we zelf organiseren.

In bijna alle gevallen zijn nieuwere versies backward compatible. Leveranciers doen hun uiterste best om bestaande code niet te breken, en worden daar uitgebreid op getest. Het verwijderen van mogelijkheden wordt meerdere maanden of jaren van tevoren aangekondigd.

Deze evolutie is een gevolg van het inzicht dat IT efficiënter werkt met Many More Much Smaller Steps, van de evolutie van een gepland waterval model naar een agile manier van werken, met continuous delivery als consequentie.

### Kwetsbaarheden

Bekommernissen omtrent cyberveiligheid hebben de versie–evolutie trend versneld.

Regelmatig worden kwetsbaarheden ontdekt in componenten en gepubliceerd (Common Vulnerabilities and Exposures – CVE databases). Wanneer software onder jouw beheer gebruik maakt van een versie van een component waarvoor een CVE bekend is, is het van het grootste belang om zo snel mogelijk over te schakelen in productie naar een versie waarin het probleem is opgelost.

Soms zijn oplossingen al voorhanden voor de CVE bekend is, en wordt bewust gewacht met het publiceren van de CVE tot een oplossing beschikbaar is om misbruik te voorkomen (responsible disclosure). Soms is die tijd er niet (zero–day kwetsbaarheden). Voor goed onderhouden componenten laat een versie waarin de kwetsbaarheid is opgelost niet langer dan enkele dagen op zich wachten.

Componenten van voldoende kwaliteit corrigeren functionele of veiligheidsgebreken voldoende snel, maar concentreren zich op de meest recente versies. In principe kunnen gebreken in oudere major versies ook hersteld worden, maar behalve bij de grootste componenten ontbreken meestal de middelen om dit op te volgen voor oudere versies. Men

focust op gebruikers die “mee” zijn. Voor oudere versies worden gebreken niet geregistreerd, niet opgevolgd, en niet verholpen.

Zo snel mogelijk updaten is een absolute noodzakelijkheid in de huidige cyberveiligheidscontext, vooral voor applicaties die publiek bereikbaar zijn, ook als authenticatie nodig is voor het gebruik, en zeker als een veiligheids– of privacy inbreuk een meer dan verwaarloosbare impact zou hebben.

### Continue updates na de initiële ontwikkeling

Wanneer software in ontwikkeling is worden de componenten continu geüpgraded. De ontwikkeltools die we gebruiken waarschuwen ons als er voor een component een nieuwe versie is, en het is een kleine moeite om die component op dat moment te upgraden.

Wanneer we echter in een periode terechtkomen waar niet frequent aan de applicatie gewerkt wordt, is het toch nodig ze up–to–date te houden. We hebben geleerd dat het veel efficiënter en kosteneffectiever is om de evolutie van de componenten die we gebruiken continu op te volgen, dan langere tijd te wachten.

Dat is onder andere zo omdat componenten onderling versie–afhankelijk zijn, en er een volgorde is waarin de updates kunnen gebeuren. Het is erg moeilijk om die volgorde maanden later te bepalen. Er is weinig reden om een bug fix of security update niet quasi onmiddellijk toe te passen. Nieuwe mogelijkheden hebben geen invloed op bestaande code. Zelfs een major nieuwe versie van een component breekt de functionaliteit van een bestaande applicatie bijna nooit. En als een nieuwe versie de functionaliteit van een bestaande applicatie zou breken, wil je dat zo snel mogelijk weten. Vroeg of laat moet je toch door deze update heen, of de component vervangen door een andere aanpak.

Daarom raden we aan om na de initiële ontwikkeling, tijdens periodes waar niet frequent aan de applicatie gewerkt wordt, toch maandelijks een update uit te voeren.

Voor elk artefact wordt dan maandelijks:

- gekeken of er nieuwe versies zijn voor de gebruikte platformen, bibliotheken, en frameworks, of updates van componenten van de stack

- worden voor elke van deze updates de release notes bekeken, om in te schatten of ze impact zouden hebben op de functionaliteit van de applicatie
- worden de updates één voor één, of in kleine groepjes
  - toegepast
  - en worden de geautomatiseerde testen uitgevoerd

Als de bestaande geautomatiseerde testen zonder probleem lukken, en de testen dekken de ganse functionaliteit, kunnen de wijzigingen zonder verdere verificatie in productie worden uitgerold. Als er toch nog problemen zouden opduiken kan de uitrol teruggerold worden. Voor uitgebreide applicaties is het beschreven proces enkel economisch haalbaar wanneer er automatische testen zijn die vertrouwen geven dat de volledige applicatie werkt wanneer ze slagen.

### Long Time Support

Platformen, frameworks met grote impact, en delen van de stack zoals het operating system, werken vaak met een Long Term Support (LTS) aanpak volgens een aangekondigd schema. De leverancier spreekt de intentie uit om een major versie tijdens een van tevoren bepaalde periode backward compatibel te onderhouden. Dit betekent dat actief wordt gewerkt om zo snel mogelijk patch versies beschikbaar te stellen die functionele of veiligheidskwesties oplossen die ontdekt werden, en dat verder erg conservatief met veranderingen wordt omgegaan, zodat de kans dat iets breekt zo klein mogelijk is. `minor` versies worden enkel uitgerold als de leverancier zich erg zeker voelt over de nieuwe functionaliteit.

Voor componenten met een LTS–schema worden enkel versies gebruikt die in de LTS periode zitten. LTS–schema's voorzien in een overgangsperiode van 6 maanden, een jaar, of langer. De LTS–periode voor een versie `v6` begint voor de LTS–periode van een vorige versie `v4` afloopt. Wanneer de toepassing uitgerold is met major LTS versie `v4` hebben we tijd om over te schakelen naar `v6` tot het einde van de LTS periode van `v4`. We doen de upgrade naar `v6` in productie niet eerder dan de start van de LTS–periode van `v6`. Deze overgang gebeurt best zo snel mogelijk na de start van de LTS–periode van `v6`, om te voorkomen dat we tegen de deadline van de LTS periode van `v4` aanlopen.

Om dat voor te bereiden is het verstandig om de automatische testen in de ontwikkelingspijplijn ook te draaien tegen de meest recente niet–LTS versies van deze platformen en operating systems. Alhoewel we weten dat `v6.1.1` geen LTS is, volgen we zo toch de evolutie van nabij op, en worden we vroeg gewezen op eventuele problemen met de komende `v6` LTS, terwijl productie nog met `v4` LTS werkt.

Op die manier wordt de periode waarbinnen een nieuwe toekomstige LTS–versie beschikbaar is, voor ze LTS wordt, gebruikt door leveranciers van bibliotheken, frameworks, andere stack–componenten, en cloud–providers, om zich voor te bereiden op de nieuwe versie. Voor componenten met belangrijke impact is het bijna steeds zo dat ze klaar zijn voor de nieuwe major LTS versie wanneer deze beschikbaar wordt. Voor kleinere componenten en in occasionele gevallen is dit nog niet onmiddellijk zo, en moeten we nog even wachten. Dit wordt lager besproken.

### Niet-vanzelfsprekende updates

Wanneer bij een update van een component toch een probleem zou opduiken, of er zijn CVE's gemeld voor componenten waarvoor nog geen oplossing beschikbaar is, kan de update niet zonder meer worden uitgevoerd. In dat geval wordt de kwestie uitgezocht. Wat is er precies aan de hand? Is de kwestie bekend, en wordt er al aan gewerkt? Er zijn een aantal redenen waarom dit kan gebeuren:

- Het is mogelijk dat de nieuwe versie van een component incompatibel is met de code. In dat geval wordt de incompatibiliteit zo snel mogelijk opgelost, en wordt alsnog naar de nieuwe versie overgeschakeld.
- Het is mogelijk dat de nieuwe versie van een component een fout bevat. In dat geval wordt de fout gemeld aan de ontwikkelaars van de component als de kwestie nog niet bekend is, of worden onze bevindingen toegevoegd aan de bekende kwestie. Eventueel wordt een oplossing voorgesteld.
- Het derde geval komt typisch voor wanneer frameworks en plug-ins voor de frameworks gebruikt worden. Plug-ins hangen vaak af van bepaalde versies van het framework waarin ze werken. Wanneer een nieuwe versie van het framework beschikbaar wordt gesteld hebben de ontwikkelaars van de plug-ins soms tijd nodig om hun werk compatibel te maken met de nieuwe versie van het framework. Tot alle

plug-ins die we gebruiken in onze code aangepast zijn om samen te werken met de nieuwe versie van het framework kunnen we het framework en de niet–compatibele plug-ins nog niet updaten. Dit is ook het geval wanneer componenten incompatibel blijken te zijn met een nieuwe LTS versie van een onderliggende component.

In de laatste gevallen is het zinnig om te wachten met de update. Gemelde fouten in nieuwe versies van componenten die goed onderhouden worden, worden meestal op enkele dagen of weken opgelost in een nieuwe versie. Goed onderhouden plug-ins worden meestal compatibel gemaakt met nieuwe versies van het framework waarin ze werken binnen enkele weken of maanden, terwijl er geen functionele of veiligheidsproblemen zijn met de vorige versie–combinaties. In dit geval wordt nota genomen van het feit dat de componenten in kwestie niet geüpdatet zijn, en waarom, bij onze code. Bij de volgende iteratie worden deze kwesties opnieuw bekeken.

Occasioneel komt het voor dat een component niet langer onderhouden wordt. Het heeft dan geen zin om verder te wachten. Dat is enkel problematisch als er in de component of zijn afhankelijkheden kwetsbaarheden ontdekt worden, of als de componenten niet meer mee evolueren met het ecosysteem waarin ze werken. In dat geval moet de component zo snel mogelijk vervangen worden door een alternatief. Voor componenten met een grote impact, die door veel partijen in de wereld gebruikt worden, ontstaat vaak een alternatief dat door andere partijen verder onderhouden zal worden. Wanneer deze alternatieven beschikbaar zijn, of beloofd, misschien na enkele weken of maanden, is het het verstandigste naar de alternatieven over te schakelen. Voor kleinere componenten is het efficiëntste pad vaak om de code van de component in onze code op te nemen, en verder zelf te onderhouden.

De vraag wordt dan hoelang we wachten voor we zulke ingreep doen.

- Als richtlijn stellen we dat we actie ondernemen na 6 maanden zonder uitzicht op oplossing, of wanneer duidelijk wordt dat een oplossing niet te verwachten is.
- In geval van een bekende kwetsbaarheid evalueren we of de kwetsbaarheid zich in onze code voordoet, en hoe groot het risico voor onze toepassing is. Is het verantwoord om 6 maanden te wachten, of moet er sneller een oplossing gerealiseerd worden?

- Wanneer een component blijkt geen versie te zullen bieden die compatibel is met een volgende LTS–versies van onderliggende componenten voor de LTS–periode van de vorige LTS–versie afloopt, moet zo snel mogelijk ingegrepen worden.

### Opstart

Het hierboven beschreven proces van continue updates gaat er natuurlijk vanuit dat de bestaande versie een maand geleden up–to–date was. Wanneer de begintoestand van het cybersysteem niet up–to–date is, moet eerst een inspanning gedaan worden om in deze situatie te geraken.

### Aandacht van regelgeving

Al de componenten die gebruikt worden in de creatie en de uitbating van een cybersysteem maken deel uit van de toeleveringsketen (supplychain). De verordening en richtlijnen van de Europese Unie en hun toepassing in de Belgische wetgeving hebben buitengewoon veel aandacht voor deze kwesties. Een supplychain attack heeft gevolgen voor zeer veel organisaties, en er zijn redelijk wat inbreuken en afgestopte gerichte pogingen bekend. Langs de andere kant is het ook echt niet zo ingrijpend om hier gepast mee om te gaan. Daarom vereist de CRA bijvoorbeeld ook dat bij elk product met een digitale component verhandeld in de EU binnenkort een Software Bill of Materials (SBOM) meegeleverd moet worden. Dat is een ingrediëntenlijst, die opsomt welke componenten in welke versie in de toepassing gebruikt worden, zodat de klant of gebruiker zelf kan verifiëren of er bekende kwetsbaarheden in zitten.

Met maken van een SBOM, en verificatie of er meer recente versies zijn, en of er kwetsbaarheden bekend zijn voor de gebruikte versies, wordt tegenwoordig door de meeste relevante platform out–of–the–box voorzien. Voor die platformen is er dan ook weinig reden om dit niet voor elke versie van een cybertoepassing die in productie gebruikt wordt regelmatig te verifiëren.

Voor de uitvoeringsomgeving (de infrastructuur, het operating system en de rest van de stack waarop de toepassing draait) is dit op dit moment nog minder vanzelfsprekend. Het is een van de intenties van de CRA om dit aan te pakken.

## Operationele veiligheid

Hierboven concentreerden we ons op de CIA&PP aspecten bij het ontwikkelen en onderhoud van cybersystemen. Daarnaast moeten we ook aandacht besteden aan de operationele omgeving van het cybersysteem.

Dezelfde redeneringen zijn van toepassing.

- Neem veiligheidsaspecten vanaf de vroegste plannen van de operationele infrastructuur en omgeving expliciet mee (secure-by-design, shift left) en maak ze inzichtelijk in een STRIDE-threatmodel.
- Pas diepteverdediging (defense–in–depth) toe. Schakel geen bestaande veiligheidsmaatregelen uit, en overdenk en activeer de veiligheidsfuncties die de infrastructuurcomponenten die je functioneel nodig hebt ondersteunen.
- Houd alle onderdelen van de operationele infrastructuur en omgeving continu up–to–date. In cloudomgevingen verzorgt de aanbieder dit doorgaans dagelijks. Bij eigen infrastructuur is dit één van de meest belastende en kostelijke opdrachten.

### Diepteverdediging

Diepteverdediging houdt onder andere in dat componenten enkel bereikbaar zijn op manieren die functioneel nodig zijn, en enkel de noodzakelijke mogelijkheden ondersteunen.

De machine waarop een webtoepassing draait reageert enkel op verkeer op poort 443 (HTTPS), en op de machine zijn enkel die OS– aspecten en derde applicaties geïnstalleerd die absoluut noodzakelijk zijn voor de toepassing (hardening). De machine waarop een database server draait reageert enkel op verkeer op de poort waarop de database server aangesproken wordt, en op de machine zijn enkel die OS–aspecten en derde applicaties geïnstalleerd die absoluut noodzakelijk zijn om de database server te laten werken. Het netwerk laat enkel verkeer toe vanuit het internet naar machines waarop webtoepassingen draaien, enkel naar de nodige poorten (netwerksegmentatie, gedemilitariseerde zone). Het netwerk laat enkel verkeer toe tussen machines en op poorten waar dat noodzakelijk is. Hiervoor wordt vaak een firewall gebruikt.

Deze en soortgelijke technieken van netwerk– en machinebeveiliging zijn ondertussen al decennia bekend en beschreven, en hun toepassing is bijna vanzelfsprekend. Een voordeel van het gebruik van cloud–componenten is dat dit werk voor ons gedaan wordt. Hardening, bijvoorbeeld, is niet vanzelfsprekend.

Componenten toevoegen die niet functioneel noodzakelijke zijn voor veiligheidsredenen is bedenkelijk. Dergelijke componenten voegen accidentele, niet–essentiële complexiteit toe aan de architectuur, en complexiteit is de grootste vijand van veiligheid. Deze technologieën zijn oorspronkelijk ontwikkeld om intrinsiek onveilige cybersystemen van vorige generaties minimaal te beschermen in een internetcontext, maar ze zijn vaak duur en generiek. Voor veilig ontworpen en gerealiseerde systemen bieden ze weinig meerwaarde. Bovendien vormen ze zelf een potentieel risico: elk toegevoegde component introduceert potentieel nieuwe kwetsbaarheden die eerder niet bestonden. De meest bekende wereldwijde cyberincidenten in de afgelopen jaren waren te wijten waren aan kwetsbaarheden in dergelijke extra componenten (bijvoorbeeld: Log4Shell in firewalls, CrowdStrike).

## Backup en recovery

Backup en recovery is een apart operationeel aandachtspunt. Als manier om om te gaan met unknown unknowns is het nodig om na te denken over de waarschijnlijkheid en impact van onvoorziene gebeurtenissen en calamiteiten. De meest algemene mitigatie hiertegen is te voorzien dat het ganse cybersysteem van nul opnieuw opgezet kan worden. Hiervoor is op zijn minst nodig dat de toepassing en de data nog beschikbaar zijn, wat geregeld wordt met backups. Hoe lang mag het duren voor het cybersysteem terug beschikbaar is? Hoeveel werk is er verloren wanneer we starten van de laatste backup? Wat is aanvaardbaar tegen welke kost? Hoe zorgen we ervoor dat dit effectief mogelijk is binnen de grenzen die we stellen?

Het is daarnaast belangrijk dat de recovery plannen regelmatig effectief uitgevoerd worden, om te verzekeren dat de backups ook inderdaad beschikbaar zijn, en de plannen aan de verwachtingen voldoen.

Een backup– en recoveryplan is een belangrijk aspect in het kader van de ISO 27000 familie van standaarden.

## Logging en observatie

Na al ons werk blijven de unknown unknowns. We hebben ons best gedaan om de known knowns en de known unknowns te mitigeren met goede ontwerpkeuzes en bijkomenden technische maatregelen, die we ook geverifieerd hebben. Maar iedereen is faalbaar.

Daarom is het belangrijk om het gedrag en gebruik van het cybersysteem in het oog te houden. Dit doen we door elk gebruik te registreren, de relevante beslissingen die het cybersysteem neemt, en hoe het daartoe komt, te registeren in logs en audit trails.

Die logs worden later gebruikt om het gedrag van het systeem te analyseren, bijvoorbeeld wanneer een fout gedrag gemeld wordt, of een inbreuk tegen de cyberveiligheid van het systeem. Wat is er net gebeurd? Welk deelaspect gedraagt zich niet zoals we bedoelden of verwachten? In een wettelijk kader zijn logs en audit trails vaak erg belangrijk, als verdediging, of bewijsmateriaal tegen derden.

Daarnaast kunnen we na een tijd de gegevens die op deze manier verzameld zijn gebruiken om de known unknowns in kaart brengen en er na een tijd known knowns van maken. Hoe vaak gebeurt een brute force inbraakpoging tegen ons systeem (tientallen keren per dag, soms continu — het ganse internet wordt continu gescand door botnets op kwetsbaarheden)? Wat is de piekbelasting van deelsystemen (zodat we voor die deelsystemen voldoende capaciteit kunnen voorzien zodat de functionaliteit beschikbaar blijft)?

Voor cybersystemen met een erg hoog risico worden de logs continu, in real–time geobserveerd door automatische systemen. Zo kunnen inbraakpogingen of potentieel misbruik ontdekt worden terwijl het bezig is, en kan onmiddellijk gereageerd worden. Wat zulke geavanceerde systemen in moderne varianten doen is, na een initiële periode, een machine learning model opstellen van de logs tijdens “normaal gebruik”. Die modellen analyseren daarna continu de nieuwe logs, en detecteren wanneer er andere dan normale gebruikspatronen voorkomen. Dit heeft een kost, en vereist een hoge expertise om op te zetten. Meer eenvoudige systemen kunnen al opgezet worden voor cybersystemen met lager risico, door handmatig criteria te bepalen voor verdachte of gevaarlijke omstandigheden. Als het cybersysteem gisteren bijvoorbeeld erg traag reageerde, en we leren uit de logs dat er erg

veel database connecties waren, kunnen we een waarschuwing opzetten via SMS wanneer het aantal database connecties boven een bepaald aantal stijgt.

## Incident Response Plan

Hoeveel voorzorgsmaatregelen we ook nemen, de kans bestaat altijd dat er iets misgaat.

Duidelijke rollen en verantwoordelijkheden zijn essentieel bij een incident. Wie coördineert de reactie? Wie beoordeelt de ernst van het incident en beslist over escalatie? Wie zorgt voor de juiste technische maatregelen om het incident in te perken?

Communicatie is een cruciale pijler. Welke partijen moeten geïnformeerd worden, wat moet er gemeld worden, en wanneer, afhankelijk van de soort en de ernst van het incident? Extern kunnen meldingsverplichtingen gelden aan toezichthouders, klanten of gebruikers, en indien nodig de bevoegde autoriteiten, zoals de Gegevensbeschermingsautoriteit bij datalekken.

Het is belangrijk om te voorkomen dat bewijsmateriaal verloren gaat. Logs vormen de basis voor onderzoek na een incident, en kunnen helpen bij het bewijzen van schuld of onschuld, en bij de verdediging tegen claims achteraf. Deze gegevens zijn echter kwetsbaar tijdens een eerste reactie.

Juist in crisissituaties is het moeilijk om helder te denken, alles in de juiste volgorde te doen, en niets belangrijks te vergeten. Een goed voorbereid script biedt houvast en zorgt ervoor dat essentiële acties tijdig en correct worden uitgevoerd, wat cruciaal is om de impact van een incident te minimaliseren. Een incident response plan is zulk een script.

Het is belangrijk dat het incident response plan opgesteld wordt tijdens rustige tijden, voor een incident zich voordoet, wanneer de geesten helder zijn. Het draagt bij aan een gestructureerde aanpak en versterkt de veerkracht van de organisatie tegen incidenten.

Met een incident response plan vult de organisatie opnieuw een belangrijk onderdeel van de ISO 27000-denkkaders in.

## Andere aspecten

In deze tekst bespraken we enkel die aspecten van de ISO 27000-denkkaders waarbij PeopleWare kan helpen.

Om uw cyberveiligheidspositie volledig te maken in het kader van de verordeningen en richtlijnen van de Europese Unie, en de toepassing ervan in de Belgische wetgeving, als die op u van toepassing zouden zijn, dient u daarbuiten ook de volgende aspecten te behandelen:

- Wie is voor wat verantwoordelijk, en wie kan uitzonderingen toestaan?
- Hoe worden medewerkers bewust gemaakt van de cybersecurity–regels van uw organisatie, en dreigingen voor henzelf en uw organisatie in de huidige maatschappelijke context (awareness)?
- Fysieke beveiliging, bijvoorbeeld toegang tot de gebouwen en data centers, en computers van de medewerkers. Wie kan er meekijken naar wat er op de schermen staat? Zijn er risico's bij brand?
- Toegangscontrole tot andere cybersystemen: Wie mag wat? Is MFA verplicht? Hoe gaan we om met een medewerker die ons verlaat? Hoe vergewissen we er ons van dat we voldoen aan de regelgeving die op ons van toepassing is?
- Als er gevoelige gegevens gedeeld worden, hoe doen we dat op een veilige manier?
- Hoe worden computers van medewerkers beveiligd? Tegen virussen, of aanvallen? Wat is het risico als een laptop gestolen wordt? Mogen medewerkers hun eigen smart phone gebruiken om met gegevens van de onderneming om te gaan?
- Is het netwerk van de organisatie beveiligd op voldoende wijze? Wat kan er mis gaan, en hoe erg zou dat kunnen zijn?
- Een oplijsting van de PII die in uw organisatie gebruikt wordt, en hoe u omgaat met de privacy van uw medewerkers, en andere personen, in uw cybersystemen.
- Wat laat u toe in verband met AI voor uw werknemers in de context van CIA&PP? Zijn er veiligheidsrisico's verbonden aan het gebruik van AI in producten of diensten die u aanbiedt?

Daarnaast is een opsomming van al uw kritische leveranciers nodig, en een beschrijving van de risico's die u met hen loopt, en wat u van hen verlangt in het kader van cybersecurity.