Ontwerp voor CA-tool (Continuous Auditing tool)

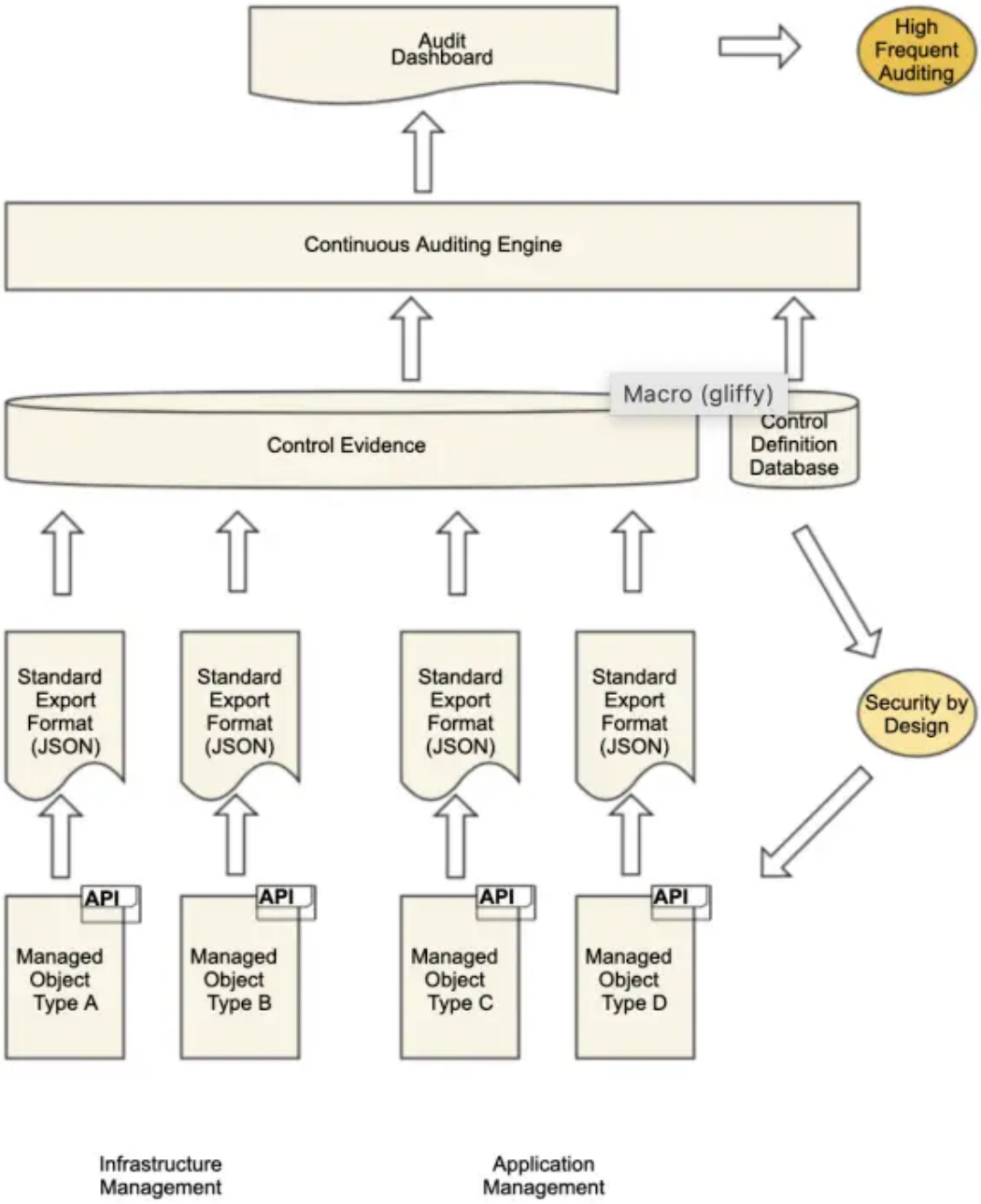
itpedia.nl/2021/06/11/ontwerp-voor-ca-tool-continuous-auditing-tool/

CE community June 11, 2021

Het concept van Continuous Auditing (CA) vereist een informatiesysteem dat moet worden gebouwd of ontworpen als elke andere IT oplossing. IT oplossingen worden tegenwoordig veelal Agile gerealiseerd of gekocht en geïntegreerd in het bestaande applicatielandschap. Dit geldt ook voor de CA-tools. Dit artikel beschrijft zowel de stappen om een **CA-tool** te realiseren als de gerelateerde vragen die beantwoord moeten worden.

Introductie CA-tool

Het concept van continuous auditing is weergegeven in figuur 1. Meer over de achtergrond van CA is te lezen in Het Concept van Continuous Auditing.



Figuur 1. Continuous Auditing Concept

Problem statement CA-tool

Om een Continuous Auditing tool (CA-tool) te implementeren, zijn er drie vraagstukken die we moeten oplossen:

- Het eerste vraagstuk omvat een aantal essentiele vragen die we moeten beantwoorden.
- Het tweede vraagstuk betreft het ontwerpen van een CA-tool (indien niet verworven).

- Ten derde het ontwikkelen, bedienen en onderhouden van de CA-tool.

CA-Tool vragen

In het eerste artikel van deze trilogie (zie Het Concept van Continuous Auditing) staan een aantal voorwaarden om de realisatie of aanschaf van een CA-tool te starten. Deze voorwaarden zijn te verwoorden in de volgende vragen :

Q01: Wat is de visie van een CA-tool?
Q02: Met welke bronnen (wet & regelgeving) of interne requirements voor informatiebeveiliging moet en we rekening houden?
Q03: Hoe deze requirements te definiëren?
Q04: Hoe vertalen we de requirements naar controls in een lokale context?
Q05: Welke informatie-assets zijn binnen het bereik te definiëren en toe te wijzen aan de controls?
Q06. Hoe extraheren, transformeren en laden (ETL) we de informatie van de assets in de CA-tool
Q07: Hoe te beslissen om de CA-tool te kopen of te bouwen?

Continuous Auditing tool requirements

Voor zowel de verwerving als de realisatie van de CA-tool dienen we een concreet beeld van het eindresultaat inclusief de CA-tool requirements te definieren. In dit artikel staat beschreven hoe we de CA-tool moeten ontwerpen en hoe we de requirements definieren. Er wordt ook een aantal CA-tool requirements aangereikt.

CA-Tool Benadering

Uiteindelijk moeten we de implementatie van de tool onderzoeken. Er zijn verschillende manieren om de CA-tool te implementeren.

CA-Tool Vragen

Er is echter geen universeel antwoord op de CA-Tool-vragen. Voor elke implementatie is afstemming op de business context nodig. De volgende antwoorden kunnen we als referentie gebruiken:

Q01. Wat is de visie van een CA-Tool?

De visie voor een CA-Tool is dat jaarlijkse audits niet voldoende zijn om aan te tonen dat de risico’s continue onder controle zijn door de dynamiek in de informatievoorziening. Door vaak diensten en producten te laten creëren en

wijzigen, bestaat het risico dat we de onderkende risico's niet onder controle hebben. Daarom moeten we de nieuwe diensten en producten voorzien van een interface om het benodigde bewijs te extraheren om in control te zijn. Een CA-tool moet dit continue monitoren.

Q02: Met welke bronnen van CA-Tool-requirements moeten we rekening houden?

Er zijn een aantal requirementsbronnen die we in de CA-tool moeten opnemen. Dit hangt af van het vereiste niveau van controle. In dit artikel veronderstelt ISO 27001:2013 als de minimale set van eisen, aangevuld met de requirements op basis van een interne risicoanalyse waarvan wordt aangenomen dat ze zijn vastgelegd in de Confidentiality Integrity & Accessibility (CIA) Matrix.

Q03: Hoe deze vereisten te definiëren?

De CA-tool requirements komen tot uiting in het gedrag van de CA-tool. In dit artikel wordt de Gherkin-taal gebruikt om het gedrag te beschijven. We kunnen ook andere formaten gebruiken. De Gherkin-taal is een krachtige beschrijvende notatie om het gewenste waarneembare gedrag van een systeem in een eenvoudige en intuïtieve syntaxis uit te drukken.

Q04: Hoe vertalen we CA-tool requirements naar controles in een lokale context?

Zo dienen requirements voor toegangsbeheer als "Multi Factor Authentication (MFA)" vertaald te worden naar een controle die past in de context van de lokale situatie. De uitvoering van de control die voldoet aan de MFA-eis kan per organisatie verschillen, maar de meting van de effectiviteit van de control kan worden gegeneraliseerd naar het percentage accounts dat voldoet aan de MFA-eis. Op deze manier kan een algemeen controlekader met gerelateerde effectiviteitsmeting worden gedefinieerd. De meting kan worden uitgedrukt in een JSON-formaat. Dit maakt het mogelijk om een API te implementeren om de benodigde informatie te verstrekken.

Q05: Hoe definiëren we de activa binnen het bereiken wijzen we die toe aan de besturingselementen?

CIA-matrix als filter voor assets

De requirements zijn beperkt tot de informatiebeveiliging requirements voor activa die worden gebruikt om informatiediensten te verlenen die worden beheerd door het Information Security Value System (ISVS). Deze scope

omvat echter alle activa met alle mogelijke risico's. Daarom moet de CIA-matrix worden gebruikt om de activa te filteren op degenen met een hoog risico en die moeten worden opgenomen in de CA-tool.

Mapping van assets

De koppeling van deze activa van de CIA-matrix aan de controles is tweeledig aangezien er twee soorten controles zijn: de externe (ISO 27001: 2013 / Wet & regelgeving) en de interne controles (tegenmaatregel voor risico's die de organisatie zelf heeft geïdentificeerd). Deze controles kunnen elkaar overlappen.

Q06: Hoe de informatie van de activa naar de CA-tool extraheren, transformeren en laden (ETL)?

De benodigde ETL-functie om informatie uit te wisselen tussen de geïmplementeerde controles en de CA-tool moet gestandaardiseerd zijn. Om de bewijsverzamelaar te maken (zie figuur 1) is er een gegevensextractie uit de managed objects. Deze informatie moet echter worden omgezet in het uniforme formaat van de collector van bewijsmateriaal. Ten slotte moeten de getransformeerde gegevens worden geladen in de database met controlebewijs.

Tegenwoordig is JSON een veelgebruikte interfacedefinitie, maar XML kan dat ook doen. Er zijn veel technieken en tools op de markt om dat op een gecontroleerde manier te bewerkstelligen.

Q07: Hoe te beslissen om de CA-tool te kopen of te bouwen?

De keuze om een CA-tool te bouwen of te kopen, hangt af van het aantal objecten dat we moeten bewaken en het percentage objecten dat een gestandaardiseerde interface heeft om informatie te extraheren die compatibel is met de aan te schaffen CA-tool. Beide keuzes hebben voor- en nadelen. Elke organisatie moet beide oplossingen in evenwicht brengen.

	Pro	Contra
Zelf ontwikkelen	· Afgestemd op de behoeften	· Kostbaar om te bouwen· Moet worden bijgewerkt voor alle innovaties· Onderhoud· Ondersteuning· Operationele kosten
Common Of The Shelf (COTS)	· Snelle implementatie· De verkoper breidt de functionaliteit in de loop van de tijd uit· CAPEX-kosten	· Moet worden geconfigureerd voor MO`s die niet passen· Moet worden opgeleid· De prijs kan hoog zijn bij grote volumes MO`s

CA-Tool Requirements

Het eerste deel van de CA-Tool-requirements is het ontwerp dat de werking van de CA-tool wereldwijd beschrijft met behulp van Agile-ontwerptechnieken. Het tweede deel zijn de vereisten die het gedrag van de CA-Tool in veel meer detail definiëren, waarbij de vereisten, acceptatiecriteria en testcases worden gecombineerd door het gebruik van de augurken-taal.

Hoe ontwerp je een CA-tool?

De stappen die in dit artikel worden gevolgd zijn:

Stap-1. Definieer de value stream voor CA-tool.
Stap-2. Definieer de use case diagrammen.
Stap-3. Definieer de use cases.

Stap 1. Definieer value stream voor CA

Een value stream is een reeks stappen om een resultaat te creëren. Meer value streams vormen samen een value system, in dit geval een CA Value System (CVS). Dit value system kan in het ISVS worden gebruikt om de value stream internal auditing van het ISVS te automatiseren.

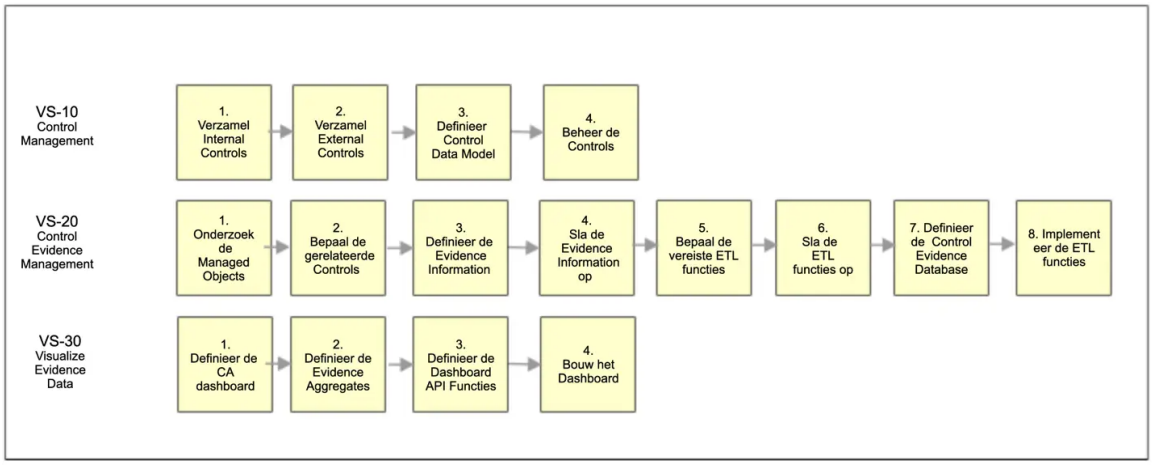
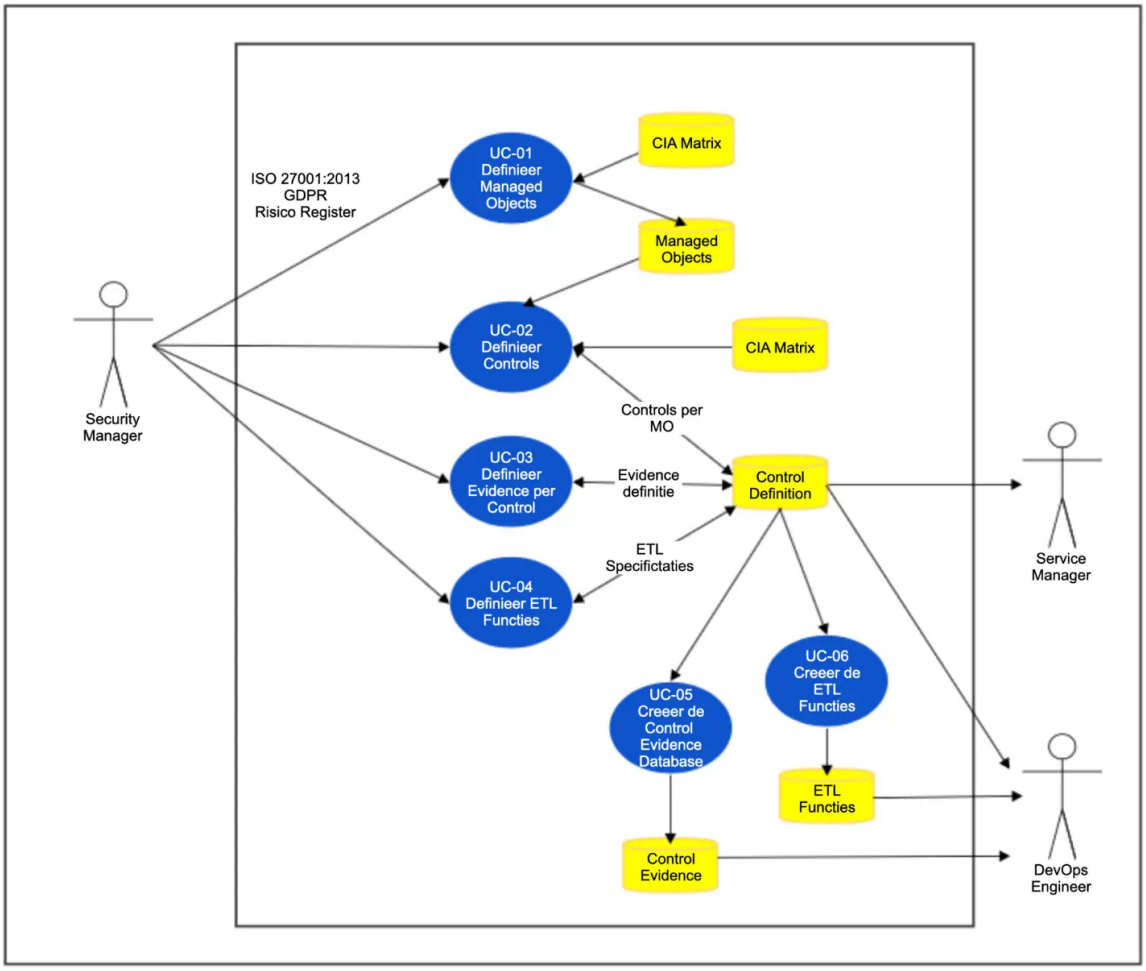


Figure 2. De value streams van het CA Value System.

De eerste value stream creëert de Control Definition Database (CDD) zoals gedefinieerd in figuur 1. Deze database bevat alle controles die automatisch moeten worden gecontroleerd. De tweede value stream definieert de datastructuur voor de Control Evidence Database. Deze database is gevuld met het bewijs van de effectiviteit van de controles door gegevens te extraheren uit de beheerde objecten die binnen het bereik vallen. De ETL-functies om dit te bereiken zijn ook gedefinieerd in deze value stream. Last but not least wordt de visualisatie bepaald en de geaggregeerde informatie berekend. De gegevens worden geëxporteerd door het gebruik van REST API`s.

Stap 2. Definieer de use case-diagrammen

Voor elke value stream moet ten minste één use case-diagram worden gedefinieerd. Een voorbeeld wordt getoond in figuur 3 voor VS-020. Hier worden de stappen vertaald naar use cases. Ook worden de actoren per use case gegeven. Dit is de tussenstap naar de definitie van de use case voor de CA-tool.



Figuur 3. De use case-diagrammen voor beheer van evidence (VS-02 value stream).

Stap 3. Definieer de use cases

Voor elk blauw ovaal van de use case-diagrammen moet een use case worden gedefinieerd. Als de use-cases vrij duidelijk zijn, kan er slechts één use-case worden gemaakt waarin de hele UCD wordt gedefinieerd. Dit bespaart tijd bij het definiëren van de use cases.

On-derwerp	**M/O**	**Beschrijving**
ID	M	UC-20
Naam	M	Control Evidence Management
Doel	M	

		Het doel van Control Evidence Management is om de evidence te verzamelen voor de gedefinieerde controles en de visualisatie van het evidence mogelijk te maken door de evidence gegevens in het vereiste formaat te exporteren
Samen vatting	M	Het verzamelen van het evidence vereist het definiëren van de assets die binnen het toepassingsgebied vallen. Voor elk asset moeten de toepasselijke controles worden geselecteerd en toegepast. Het evidence voor de effectiviteit van de zeggenschap wordt bepaald door een waardering op de assets. Door de evidence in een centrale database te verzamelen, kan de effectiviteit van de controles worden gevisualiseerd. De evidence wordt waar nodig verzameld, b.v. de gemiddelde prestatie van een maand berekend door dagelijkse metingen.
Pre-condit-ie(s)	M	Voorwaarde voor deze use case is dat: · Er is een CIA-matrix met het geclassificeerde managed object · De interne en externe controles worden gedefinieerd en opgeslagen in de Control Definition Database.
Res-ultaten in geval van succes s	M	De randvoorwaarde voor deze use case is dat: · De scope van de beheerde objecten die door de CA-tool moeten worden beheerd, wordt gedefinieerd. · De controles per managed objecten zijn bekend. · Het evidence formaat wordt geïdentificeerd en gedefinieerd in een JSON-formaat. · De Control Evidence Database wordt gecreëerd die de opslag van het bewijs mogelijk maakt. · Voor elk evidence formaat wordt de ETL-functie gedefinieerd en geïmplementeerd.

<table>
<tr><td></td><td></td><td>· De Continuous Auditing Engine kan de ETL-functies aanroepen om de evidence uit de managed objecten te extraheren, de evidence omzetten in het formaat van de Control Evidence Database en de evidence laden.

· Het evidence formaat worden waar nodig geag-gregeerd.</td></tr>
<tr><td>Res-ultaten in geval van falen</td><td>M</td><td>· De ETL-functies kunnen de meting niet uitvoeren.

· De geëxtraheerde informatie kan niet worden gecon-verteerd naar het formaat van de Control Evidence Database.</td></tr>
<tr><td>Per-form-ance</td><td>O</td><td>De prestatiecriteria die van toepassing zijn op deze use case.

LT = 15 min
PT = 15 min
%A/C = 99%

Deze tellers zijn gebaseerd op de tijd die nodig is om deze use case uit te voeren in de CI / CD Secure-pipepline voor één sprint. De eerste uitvoering kost veel meer tijd.</td></tr>
<tr><td>Fre-quentie</td><td>M</td><td>Deze use case wordt</td></tr>
<tr><td>Actoren</td><td>M</td><td>De security manager, de servicemanager en de DevOps Engineer zijn nodig om deze use case uit te voeren.</td></tr>
<tr><td>Trigger</td><td>M</td><td>Tijdtrigger, vier keer per dag.</td></tr>
<tr><td>Scen-ario A (in tekst)</td><td>M</td><td>
<table>
<tr><th>Step</th><th>Actor</th><th>Value Chain Activ-iteit</th><th>Stap naam</th><th>Activieit</th></tr>
</table>
</td></tr>
</table>

			1	Security Manager	Engagement	Defineer Managed Objects	Selecteer de managed objects die moeten worden opgenomen. Alle assets worden normaal geregistreerd in het asset register of de configuratiemanagementdatabase. Dit omvat echter alle assets. De CA-tool wordt alleen gebruikt voor de informatiebeveiligingsrisico gerelateerde assets. Daarom moet de CMDB worden gefilterd op basis van de CIA-matrix waarin de assets worden geclassificeerd op hun kwetsbaarheid.
			2	Security Manager	Engagement	Definieer controls per Managed Objects	Definieer het filter voor de selectie van bedieningselementen die moeten worden gevisualiseerd. Dit filter is gebaseerd op een aantal criteria zoals:

							· Kritische controles op basis van toepasselijke wet- en regelgeving · Kritische controles op basis van het risicoregister Op basis van dit filter kunnen de controles uit de CIA-matrix (interne controles) en de ISO 27001: 2013 (externe controles) worden geselecteerd. Deze informatie is normaal gesproken beschikbaar in de Statement of Applicability. De geselecteerde controles moeten worden toegewezen aan de managed objects, aangezien niet alle controls van toepassing zijn op alle managed objects.
			3	Security Manager	Design & Transition	Definieer evidence per control	De effectiviteit van elk control vereist evidence. Het evidence

						wordt verkregen door een meting uit te voeren. De output van de meting heeft zijn eigen unieke bewijsformaat.
			4	De-vOps En-gineer	Design & Transition	Definieer ETL functies	De meting moet worden geïmple-menteerd door de definitie van de extractie van de informatie, de transformatie naar het formaat van de Control Evidence Data-base om te worden geladen.
			5	De-vOps En-gineer	Design & Transition	Creeer de Con-trol Evid-ence Databas e	De structuur van de Control Evid-ence Database moet voldoen aan de inform-atiebehoefte van het dashboard. De database-structuur moet overeenkomen met alle evidence formats.
			6	De-vOps En-gineer	Deliver and Support	Creeer the ETL functies	Het dashboard moet worden gevuld met be-wijsinformatie. De populatie van de Control Evidence

<table>
<tr><td></td><td></td><td colspan="5">Database kan worden geïmplementeerd met microservices die het extraheren, transformeren en laden van de vereiste informatie uitvoeren.</td></tr>
<tr><td>Variatie(s) op het scenario</td><td>O</td><td colspan="5"><table><tr><th>Stap</th><th>Actor</th><th>Value Chain Activiteit</th><th>Stap naam</th><th>Activiteit</th></tr><tr><td></td><td></td><td></td><td></td><td></td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr></table></td></tr>
<tr><td>Open vragen</td><td>O</td><td colspan="5">Welke techniek wordt gebruikt om de gegevens uit de Control Evidence Database te halen om de gegevens op het dashboard te visualiseren?</td></tr>
<tr><td>Planning</td><td>O</td><td colspan="5">Q1 2021</td></tr>
<tr><td>Risico</td><td>O</td><td colspan="5">Niet alle assets kunnen worden gemeten</td></tr>
<tr><td>Impact</td><td>O</td><td colspan="5">Nieuwe assets kunnen alleen worden gepromoot naar de productieomgeving als de CA-tool wordt aangepast.</td></tr>
<tr><td>Prioriteit</td><td>O</td><td colspan="5">De prioriteit van deze use case is hoog vanwege de risico’s die moeten worden beheerst en het handmatige werk dat ermee gemoeid is.</td></tr>
<tr><td>Super Use case(s)</td><td>O</td><td colspan="5">UC-10.</td></tr>
</table>

Interface	O	De Control Evidence Database heeft geen GUI, maar de ETL-functies worden gedefinieerd door de JSON-interfaces. Ook de interface met het dashboard moet in deze use case worden gedefinieerd.
Relaties	O	De use case 10 en 30 zijn direct verbonden met deze use case.
Jira Ticket		J100

Hoe de requirements te ontwerpen?

De requirements van de CA-Tool worden beschreven in het Gherkin-formaat. In tabel 1 staan de vereisten in dit formaat voor stap 4 van UC-20.

Feature : Definieer ETL-functies voor de CA-tool
As een Security Manager
I want evidence informatie extraheren uit de assets die de controles implementeren
So that Ik de effectiviteit van de gerelateerde controles kan visualiseren.

#S1.
Scenario: Extraheer informatie om te bewijzen dat de MFA-functie wordt toegepast voor alle gebruikers van de applicatie XYZ
Given het feit dat Multi Factor Authentication vereist is voor alle gebruikers van applicatie XYZ
When Ik de gebruikerstabel in de datbase van applicatie XYZ opvraag
And de gebruikers selecteer met de MFA-vlag ingesteld op inactief
Then Ik ontvang geen gebruikersaccounts

#Verwachte outputformat :

{
”Account nr“: “<Nr>”,
”MFA Flag”: “<No>”
}

Voorbeelden van CA-tool ontwerp

Om de CA-tool een concreter gezicht te geven worden hieronder een aantal tips gegeven.

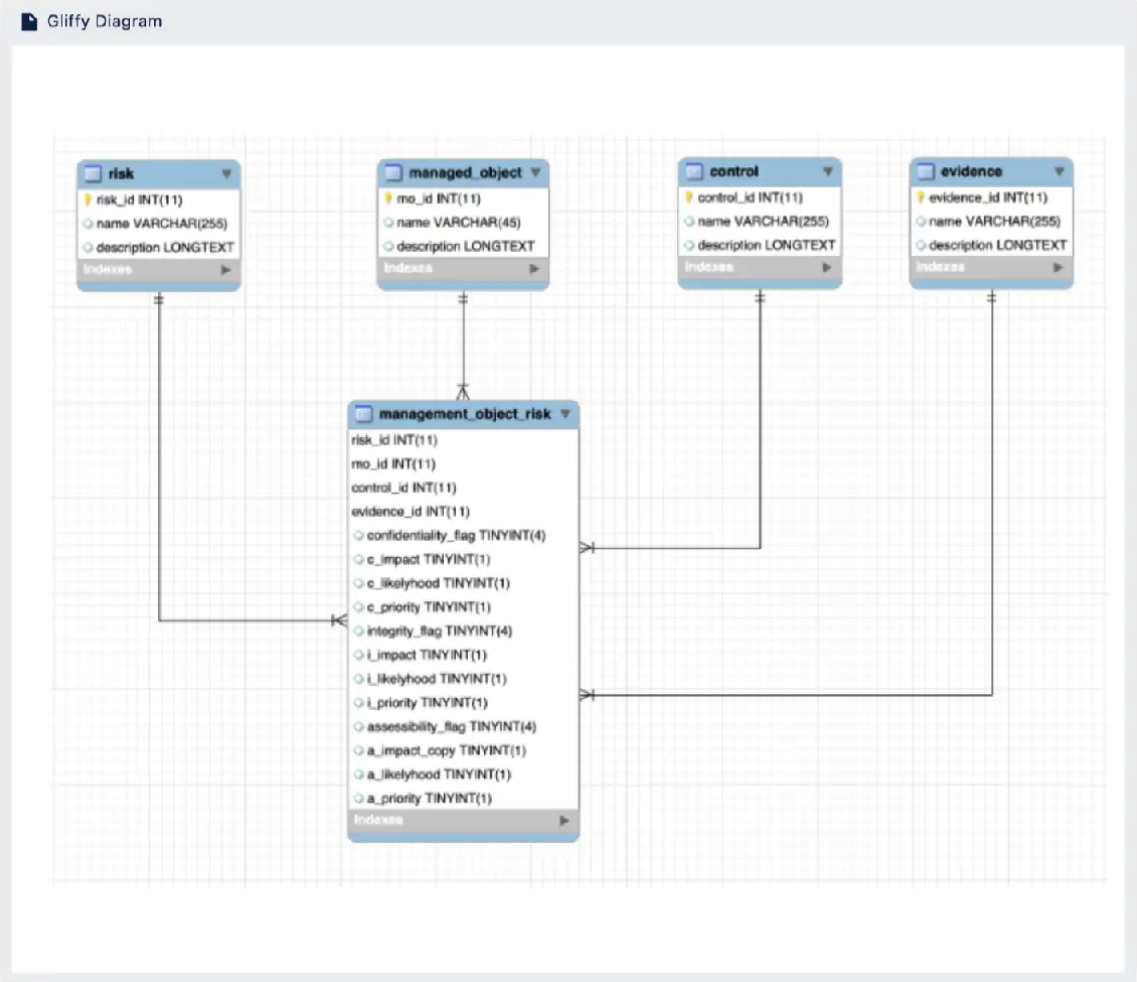
GUI

De GUI van de CA-tool verschilt niet veel van elk ander monitor dashboard. Dit betekent dat het moet kunnen:

- Visualiseer de algemene status van de informatiebeveiligingsdoelen in termen van vertrouwelijkheid, integriteit en toegankelijkheid.
- Drilling-down functionaliteit die het mogelijk maakt om details van uitzonderingen, waarschuwingen of informatieve gegevens te vinden.
- Differentiatieweergaven om de algehele kwaliteit te zien, zoals:
  – ISVS Value Streams
  – Events per value stream
  – Assets
  – Events per assets
  – People
  – Events per type van users

Data model

Figuur 4 toont een voorbeeld van het CA-datamodel. Het entiteitstype `risico` vertegenwoordigt de risico`s. Een risico hoeft echter niet voor elk beheerd object van toepassing te zijn. Ook heeft een risico niet dezelfde kenmerken voor alle managed objects waaraan het risico is blootgesteld. Daarom bepaalt de combinatie van een risico en het managed object de basis voor de CIA-rating en de daarmee samenhangende beheersing en bewijsvoering.



Figuur 4. CA-tool datamodel.

CA-Tool Benadering

Best practices voor de implementatie van de CA-tool zijn:

- Zorg ervoor dat de reguliere handmatige operationele planning voor het controleren van de bedieningselementen is gedefinieerd en voorzien van versiebeheer.
  o Dit versnelt het ontwerp van de CA-tool en het formaliseren van de eisen.
- Definieer een roadmap voor alle ontworpen functionaliteit op basis van Epics.
  o Epics zijn brokken van een kwart van de tijd.
  o Elke Epic is een Minimal Viable Product (MVP) dat waarde toevoegt aan de organisatie.
  o Volg een incrementele aanpak en besef eerst de belangrijkste en meest tijdrovende.
  o Gebruik de 80/20-regel: realiseer 80% van de controles in 20% in de tijd en laat het moeilijkste om als laatste te bouwen.

- Stem het ontwerp en de requirements af op de tools op de markt (Governance Risk Compliance) en neem eventueel een beslissing over:
  o Epics zijn brokken van een kwart van de tijd.
  o Elke Epic is een Minimal Viable Product (MVP) dat waarde toevoegt aan de organisatie.
  o Volg een incrementele aanpak en besef eerst de belangrijkste en meest tijdrovende.
  o Gebruik de 80/20-regel: realiseer 80% van de controles in 20% in de tijd en laat het moeilijkste om als laatste te bouwen.
- Commercial Of The Shelf.
  o Waar Agile de beste oplossing is voor zelfgemaakte CA-Tools, kan een COTS CA-Tool geïmplementeerd worden met behulp van een installer.

Voorbeelden van CA-Tools op de markt

- Idea.
- ACL.
- Bwise.
- ARIS.
- SAP GRC.
- Oracle GRC.
- Approva Bizrights.
- Synaxion.
- Oversight.

Follow up reading

De informatie van het eerste artikel Het Concept van Continuous Auditing en dit artikel zal gebruikt worden om een MVP voor CA tooling om cloud services te realiseren.

De Continuous Auditing (CA) Guild is empowered door Jan-Willem Hordijk, CTO van Plint AB, een Zweedse localisatie-organisatie. Deze publicatie is mogelijk gemaakt door Bart de Best (www.dbmetrics.nl), Dennis Boersen (Argis IT Consultants), Freek de Cloet (smartdocuments), Jan-Willem Hordijk (Plint AB), Louis van Hemmen (www.bitall.nl) Niels Talens – www.nielstalens.nl en Willem Kok (Argis IT Consultants).

Discussieer mee op LinkedIn.