De (harde) praktijk van role engineering

Ing. P. Mienes RE

Bij het implementeren van autorisatiebeheer op basis van het RBAC-model worden role engineering en role mining vaak door elkaar gebruikt, zonder te onderkennen dat het begrippen van een verschillende orde zijn. Dit artikel geeft aan hoe een organisatie op basis van role engineering kan komen tot een transparante en goed onderhoudbare registratie van medewerkers, rollen en permissies, en gaat in op een potentieel risico van role mining. Hierbij wordt een door KPMG IRM in de praktijk beproefde aanpak geschetst voor het bepalen van rollen en de bijbehorende permissies.

Inleiding

Role Based Access Control (RBAC) staat volop in de belangstelling, niet in de laatste plaats door de eisen die in het kader van bijvoorbeeld SOX en Basel II worden gesteld aan de effectiviteit en transparantie van het autorisatiebeheer (zie [Koor04] voor de resultaten van een onderzoek terzake). Een andere reden voor het groeiende aantal RBAC-implementaties is dat enkele ondersteunende autorisatiebeheertools inmiddels een zodanig volwassenheidsniveau hebben bereikt dat ze het autorisatiebeheer op basis van RBAC goed ondersteunen, en door geautomatiseerde provisioning (zie figuur 1 en de toelichting van gehanteerde termen) wezenlijke efficiencyverbeteringen mogelijk maken. N.B. Voor de lezer die nog niet bekend is met RBAC, bieden [Ferr03], [Mien03a] of [Mien03b] een introductie.

Maar voordat het zover is, voordat de RBAC-vruchten kunnen worden geplukt, zal in de regel een intensief traject moeten worden doorlopen om de rollen en bijbehorende permissies (= toegangsrechten = autorisaties) te bepalen. Organisaties die (up-to-date!) autorisatiematrices hebben, zullen dit traject *vele malen sneller* en met *veel minder inspanning* doorlopen dan organisaties waar inzicht in de benodigde autorisaties goeddeels ontbreekt. In de laatste situatie moet in feite tientallen jaren achterstallig onderhoud worden ingehaald, door het vaststellen van autorisatiematrices. 'Achterstallig onderhoud' omdat de organisatie voor het beheersen van het autorisatiebeheerproces natuurlijk eigenlijk altijd al up-to-date autorisatiematrices had moeten hebben.

Meestal echter ontbreken deze autorisatiematrices of zijn ze niet actueel en ligt niet vast welke medewerker welke rol vervult, en welke permissies hij/zij op basis van deze rol zou moeten hebben. Dit is in een notendop de las-

Ing. P. Mienes RE
is senior IT-consultant/auditor bij KPMG Information Risk Management en is intensief betrokken bij de logische toegangsbeveiliging en het systeembeheer in grote organisaties. Voor KPMG participeerde hij onder andere in het Role Based Access Control-project van EEMA, en de RBAC-werkgroep van het Platform Informatiebeveiliging.

mienes.pieter@kpmg.nl

tigste, en meest tijdrovende vraag die tijdens de implementatie van RBAC moet worden beantwoord. In de beschikbare literatuur over RBAC worden hiervoor nauwelijks handreikingen gedaan.

Dit artikel gaat in op een in de praktijk toegepaste methode om rollen te bepalen en de bij rollen behorende permissies. Deze methode is één element uit de KPMG IRM programma-aanpak 'Identity & Access Management (I AM) in control' (zie [Koor04]). De ambitie gaat daarbij verder dan het 'simpelweg' in rollen onderbrengen van permissies: de ambitie is om het RBAC-model transparant en vooral ook goed onderhoudbaar te maken voor de organisatie. Achtereenvolgens komen aan de orde:
- ontwerpeisen;
- bepalen van rollen;
- bepalen van permissies per rol;
- voorbeeld van role engineering.

Ontwerpeisen

Medio 2005 heeft het Platform Informatiebeveiliging in de studie 'Role Based Access' een normenkader uitgebracht voor het autorisatiebeheerproces op basis van RBAC. Eén van de normen hierin luidt als volgt:

'De organisatie moet voor de inrichting van het RBAC-model richtlijnen hanteren voor het bepalen van rollen, waarbij minimaal invulling is gegeven aan:
- *...*
- *de soorten rollen die worden gebruikt:*
  – *functie-/procesgerelateerd;*
  – *taakgerelateerd;*
  – *projectgerelateerd;*
  – *organisatiegerelateerd;*
  – *geografisch gerelateerd.*
- *de rollenstructuur, waarbij minimaal aandacht besteed is aan:*
  – *het juiste gebruik van rollen;*
  – *de introductie van subrollen, connectorrollen of andere rolhiërarchieën;*
  – *het automatisch 'erven' van rollen, subrollen of connectorrollen;*
- *...'*

Vaak moet tientallen jaren achterstallig onderhoud worden ingehaald

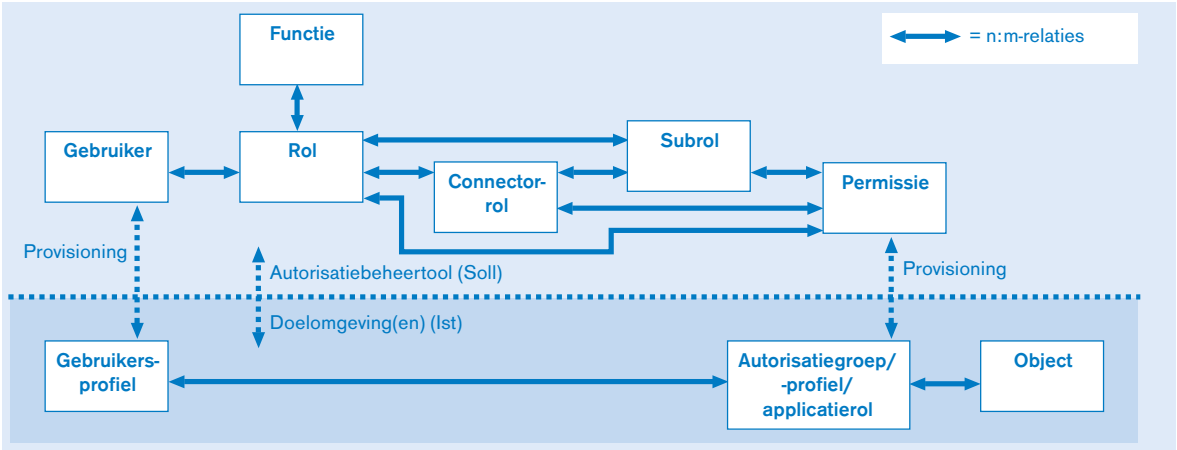
De hier genoemde elementen van de norm hebben alles te maken met de transparantie en onderhoudbaarheid van het RBAC-model: eisen die zowel van buiten af worden gesteld, als ook (vaak impliciet) door de eigen (beheer)organisatie. Het onderscheiden van permissies naar hun aard, en het op basis daarvan koppelen van die permissies aan de juiste soorten rollen vormt dé basis voor een transparant en onderhoudbaar model, dat bovendien maximaal ongevoelig is voor wijzigingen in de processen, organisatie en locaties.

Het in figuur 1 weergegeven model biedt voldoende aangrijpingspunten voor een transparante en goed onderhoudbare implementatie van autorisaties. Dit model biedt in beperkte (maar wel overzichtelijke) mate rolhiërarchieën, en is als zodanig in de meeste ondersteunende software onder te brengen (de implementatie van volledig flexibele rol-hiërarchieën ([Ferr03]) kan hier naar behoefte in worden geïncorporeerd).

Het artikel richt zich op het ontwikkelen van rollen (inclusief connectorrollen en subrollen) en het leggen van de relaties met permissies.

Bepalen van rollen

Wat de eenvoudigste stap lijkt te zijn, blijkt in de praktijk vaak nog enige haken en ogen te hebben. Als we een manager vragen naar de rol van de medewerker, dan zullen we eerst duidelijk moeten aangeven wat we verstaan onder een rol en wat het verschil is met een functie.



Figuur 1. Conceptueel RBAC-model.

Ongetwijfeld ligt per medewerker de functie vast in het personeelssysteem, maar deze heeft primair een betekenis in het functiehuis mede ten behoeve van salariëring. Zo komt het voor dat bijvoorbeeld drie medewerkers met verschillende functies allen dezelfde rol vervullen (zie figuur 2), en dus allen dezelfde permissies nodig hebben; slechts om salaris- of carrièretechnische redenen hebben zij verschillende functies! Andersom komt het ook voor (zie figuur 3): medewerkers met verschillende rollen kunnen dezelfde functie hebben.

Functie	Rol
Claimbehandelaar 1	Claimbehandelaar
Claimbehandelaar 2	Claimbehandelaar
Claimbehandelaar 3	Claimbehandelaar

Figuur 2. Verschillende functie, dezelfde rol.

Functie	Rol
Commies-A	Baliemedewerker
Commies-A	Secretariaatsmedewerker
Commies-A	Teamleider postkamer

Figuur 3. Dezelfde functie, verschillende rol.

Of we de functies uit het personeelsinformatiesysteem kunnen gebruiken als rollen, hangt dus in belangrijke mate af van de manier waarop het lijnmanagement en personeelszaken met de functies omgaan. Een gesprek met personeelszaken is daarom een eerste goede stap. In de regel is het ook verstandig om vroeg in het traject op zoek te gaan naar de ontwerpers van de bedrijfsprocessen: zij zouden in staat moeten zijn aan te geven welke rollen onderscheiden worden in het proces. Het beschikbaar hebben van zo'n overzicht van rollen, zoals deze door de ontwerper of kenner van het bedrijfsproces zijn gedefinieerd, kan vooral in grotere organisaties met decentrale kantoren erg waardevol zijn: decentraal kan immers een eigen invulling zijn gegeven aan de (namen van) rollen. Het overzicht kan dan helpen om rollen te uniformeren over de decentrale kantoren heen.

Het doel van het gesprek met de lijnmanager is om te komen tot een overzicht waarbij per medewerker is aangegeven welke rol(len) de medewerker heeft. We richten ons in deze fase primair op de functie- of procesgerelateerde rollen; in een latere fase vindt de differentiatie plaats naar andere soorten rollen (organisatie-, taak-, project- of geografisch gerelateerd).

Gehanteerde termen

- Gebruiker – definitie van een gebruiker in het autorisatiebeheertool; gebruikers kunnen zijn onder andere vaste medewerkers, inhuur, derde partijen, niet-persoonsgebonden gebruikersdefinities; via provisioning leidt de definitie tot een gebruikersprofiel in één of meer doelsystemen.
- Gebruikersprofiel – definitie van de gebruiker in het doelsysteem (bijvoorbeeld account, user-id).
- Autorisatiegroep/autorisatieprofiel – (groepering van) toegangsrecht(en) op object(en) in één doelsysteem, nodig om een rol of (deel)taak te kunnen vervullen (bijvoorbeeld Unix- of Windows-groep, (groepering van) autorisatieprofiel(en) of rollen in een applicatie).
- Permissie – afbeelding in het autorisatiebeheertool van één concrete autorisatiegroep/autorisatieprofiel/applicatierol in het doelsysteem.
- Object – datgene waarop toegangsrechten worden verkregen (bijvoorbeeld gegevens(bestand), directory, applicatie, scherm, commando, share).
- Functie – het dienstverband van de gebruiker/medewerker is in de regel gebaseerd op een bepaalde functie; in het autorisatiebeheertool wordt het begrip slechts gebruikt om de gerelateerde rollen te groeperen.
- Rol – verzameling permissies, nodig om de rol te kunnen vervullen.
- Subrol – verzameling van (bijvoorbeeld taakgebonden) permissies (eventueel in verschillende doelsystemen).
- Connectorrol – verzameling van permissies/subrollen die gemeenschappelijk zijn voor meerdere rollen.
- Soll – de invulling van het model die bepaalt wie welke autorisaties zou *moeten* hebben.
- Ist – de administraties in de doelsystemen bepalen hoe het in werkelijkheid *is* gesteld met de autorisaties.
- Provisioning – (half-)automatische koppeling tussen het autorisatiebeheertool en de doelomgeving(en), die ervoor zorgt dat in de doelomgeving(en) gebruikersprofielen worden gecreëerd/verwijderd, en koppelingen met autorisatiegroepen/autorisatieprofielen worden aangebracht/verwijderd.

In deze fase is het inzetten van role engineering- en role mining-technieken nog niet zo zinvol: het vaststellen van de rollen kan veelal het beste top-down worden uitgevoerd zoals hierboven geschetst.

Bepalen van permissies per rol

In het ideale geval beschikt de organisatie reeds over autorisatiematrices die aangeven welke autorisaties nodig zijn bij een bepaalde rol. De realiteit is meestal dat organisaties die RBAC willen implementeren juist niet beschikken over dit inzicht.

Zoals aangegeven in [Mien03a] en [Mien03b] is het niet praktisch om top-down de bij een rol behorende permissies te bepalen. Hier past de bottom-up benadering, waarbij gepoogd wordt om patronen te herkennen in de bestaande autorisaties: role engineering.

In deze paragraaf worden de uitgangspunten en werkwijze van role engineering besproken, aan het einde van dit artikel wordt een praktisch voorbeeld getoond.

De uitgangssituatie voor de hier besproken werkwijze is als volgt:
- Van elke medewerker is de rol bekend (op basis van het interview met de lijnmanager; zie vorige paragraaf).
- Van elke medewerker is bekend wat de gebruikersprofielen (accounts/user-id's) zijn op de verschillende doelsystemen.
- Van elke medewerker is bekend welke autorisaties deze heeft op de verschillende doelsystemen.
- De genoemde gegevens zijn beschikbaar in een tool dat ons ondersteunt bij de role engineering.

Het tweede punt kan in sommige organisaties een project op zich zijn. Het matchen van gebruikersprofielen aan medewerkers kan vooral lastig zijn als in het verleden bij de profielen in de doelsystemen geen identificerende gegevens zijn opgenomen (zoals een personeelsnummer of e-mailadres), en bovendien een adequate registratie van uitgegeven gebruikersprofielen gerelateerd aan medewerkers ontbreekt.

De methode voor role engineering kent een aantal stappen:
1. vaststellen van de functionele betekenis van de autorisatiegroepen en autorisatieprofielen;
2. inzichtelijk maken van de huidige autorisaties;
3. bespreken van de huidige autorisaties;
4. autorisaties in het tool opnemen als permissies en deze koppelen aan rollen.

Stap 1

Autorisatiegroepen en autorisatieprofielen hebben vaak een cryptische omschrijving, en een toelichting op die omschrijving ontbreekt veelal. Zij worden in het autorisatiebeheertool één op één afgebeeld als permissies, en vormen het grensvlak tussen het tactisch autorisatiemanagement (rollenbeheer) en de technische implementatie van autorisaties. Uit beheersbaarheidsoogpunt is het van belang om op dit grensvlak de vertaling te maken van de techniek naar de functionele betekenis van de autorisaties. Dit doen we door bij elke permissie de functionele betekenis van die permissie als commentaar op te nemen. Bijvoorbeeld: 'Openen, wijzigen en sluiten van problem tickets', of 'Fiatteren betaaltransacties'. De functionele omschrijvingen worden geleverd door de technische of functionele beheerders, en geven in voor rollenbeheerders en lijnmanagers begrijpelijke taal weer wat een gebruiker (extra) kan indien de autorisatie wordt toegekend.

Het matchen van gebruikersprofielen aan gebruikers is soms een project op zich

Stap 2

We laten het tool een matrix genereren (bijvoorbeeld per kostenplaats of andere organisatorische eenheid) met in de rijen de autorisaties per doelsysteem, en in de kolommen de medewerkers, gegroepeerd per rol. Zie figuur 4.

Het groeperen van medewerkers met dezelfde rol helpt ons bij het analyseren van de patronen en het signaleren van verschillen in autorisaties tussen medewerkers met dezelfde rol. Op de markt zijn role mining-producten beschikbaar die kunnen ondersteunen bij deze analyse; voorbeelden zijn Sage Discovery van Eurekify en SAM Role Miner van Beta Systems.

Kostenplaats XYZ		Medewerker	Medew. P	Medew. W	Medew. D	Medew. H	Medew. R	Medew, C	Medew. I	Medew. B
Doelsysteem	Autorisatie	Rol	Rol A	Rol A	Rol B	Rol B	Rol B	Rol C	Rol C	Rol D
Sys1	Aut123		X	X						
Sys1	Aut468						X	X	X	X
Sys3	Aut654				X	X	X	X	X	X
Sys3	Aut854							X		X
Sys4	Aut014		X	X				X	X	X
Sys7	AutCDE		X	X		X	X	X	X	X
Sys7	AutKLM								X	
Sys7	AutRST		X	X	X	X	X	X	X	X

Figuur 4. Matrix met huidige autorisaties.

Stap 3

De matrix en de analyse van de patronen vormen de basis voor gesprekken om per rol de benodigde permissies te bepalen. Afhankelijk van de organisatie kunnen deze gesprekken gevoerd worden met lijnmanagers (als vaak eerstverantwoordelijken voor de aan hun medewerkers uitgegeven autorisaties), autorisatiebeheerders, functioneel beheerders, technisch beheerders, database administrators, etc.

Door de gesprekken proberen we vast te stellen:
- welke autorisaties de medewerkers werkelijk nodig hebben, gegeven hun rol(len);
- welke autorisaties (dus) overbodig zijn bij sommigen;
- welke autorisaties (dus) ontbreken bij sommigen (die de autorisatie mogelijk niet gemist hebben);
- welke autorisatie (vrijwel) altijd in combinatie met een andere autorisatie wordt uitgegeven;

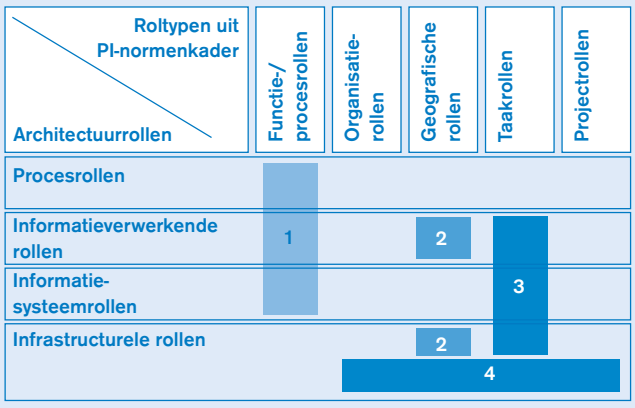
• van elke (set) autorisatie(s): wat is de *werkelijke* reden voor deze autorisatie(s)? Is het werkelijk vanwege de functie- of procesgerelateerde rol die de medewerker heeft, of is de autorisatie gerelateerd aan de organieke plek, de geografische locatie of aan een project (of ander afdelingsoverstijgend gremium)? Of heeft de autorisatie met geen van de hiervoor genoemde aspecten een directe relatie? In dit geval noemen we het een taakgebaseerde autorisatie; we zien deze vooral bij delegatie van taken, of bij het beperkt beschikbaar stellen van toegang/licenties uit oogpunt van kosten.

Role mining sec leidt slechts tot het (centraal) inzichtelijk maken van de chaos en draagt zeer beperkt bij aan de transparantie en beheersbaarheid

Het laatste punt – de werkelijke redenen voor autorisaties – dreigt bij role mining onderbelicht te blijven of zelfs niet aan de orde te komen: niets is namelijk 'eenvoudiger' dan het in één rol onderbrengen van alle permissies die nodig zijn voor die rol. Als alle benodigde permissies in één rol worden gevoegd, zonder daarbij groeperingen of hiërarchieën aan te brengen, dan is er met het RBAC-model slechts één ding bereikt: de chaos is (centraal) inzichtelijk gemaakt, maar de transparantie en onderhoudbaarheid zijn niet (significant) verbeterd.

Stap 4

Na het bespreken en duiden van de autorisaties kunnen deze als permissies in het tool worden vastgelegd (inclusief de functionele beschrijvingen), en aan de juiste soorten rollen worden gekoppeld.



Figuur 5. Raakvlakken tussen architectuurrollen en roltypen uit het PI-normenkader.

Uitgaande van de eerder gestelde ontwerpeisen hanteren we voor role engineering de volgende richtlijnen:

1. Rollen moeten zo 'schoon' mogelijk worden gehouden, dat wil zeggen dat er enkel permissies aan worden gekoppeld die werkelijk een directe relatie hebben met de functie-/procesrol, het project, de plek in de organisatie of de geografische locatie. De functie-/procesgerelateerde rollen die op deze manier ontstaan overleven in de regel elke reorganisatie, verhuizing en project, hetgeen leidt tot minder onderhoud in het RBAC-model.
2. Een subrol wordt geïntroduceerd als permissies een logische samenhang hebben, dat wil zeggen (vrijwel) altijd in combinatie met elkaar zullen worden gebruikt.
3. Een connectorrol wordt geïntroduceerd als er een significante overlap is in de aan verschillende rollen gekoppelde subrollen of permissies; de overlappende subrollen of permissies worden gecombineerd in de connectorrol.

Door het hanteren van deze richtlijnen worden de transparantie en de onderhoudbaarheid van het RBAC-model bevorderd. In de laatste paragraaf wordt role engineering nader toegelicht in een praktisch voorbeeld.

Het geschetste model en de besproken methode zijn niet heilig: vanzelfsprekend kunnen (en soms: moeten) deze worden aangepast op basis van bijvoorbeeld het informatiebeveiligingsbeleid, beheerstructuren en verdeling van taken, verantwoordelijkheden en bevoegdheden, outsourcingcontracten, tooling, bedrijfscultuur, etc.

Vergelijking met 'architectuurrollen'

In [Hofm05] wordt een aardig beeld geschetst van de problematiek rond role engineering, waarbij architectuur gehanteerd wordt als structureringsmechanisme voor role engineering. Hierbij wordt onderscheid gemaakt tussen procesrollen, informatieverwerkende rollen, informatiesysteemrollen en infrastructurele rollen. Hoe verhoudt die structurering zich tot de structurering naar de soorten rollen uit het PI-normenkader zoals dat in dit artikel wordt besproken? In figuur 5 zijn de raakvlakken weergegeven.

Toelichting op figuur 5:

1. Autorisaties die nodig zijn voor het uitvoeren van een functie- of procesgerelateerde rol bevinden zich meestal in informatiesystemen en achterliggende gegevensbestanden; minder vaak zijn hiervoor autorisaties nodig in de infrastructuur.
2. Afhankelijk van de geografische locatie kunnen specifieke autorisaties nodig zijn op gegevensobjecten (bijvoorbeeld klanten behorende tot de regio / het district) of in de infrastructuur (bijvoorbeeld fysieke toegangsbeveiliging).

3. Specifieke taken (en de benodigde autorisaties) vinden zelden hun oorsprong in het proces, maar kunnen wel gerelateerd zijn aan gegevensobjecten (bijvoorbeeld fiatteringsgegevens), informatiesystemen (bijvoorbeeld fiatteringstransacties) of infrastructuur (bijvoorbeeld het gebruik van MS-Project).
4. Autorisaties in de infrastructuur zijn vaak gerelateerd aan (naast het onder 2 en 3 behandelde) de plek in de organisatie (bijvoorbeeld toegang tot afdelingsmap) of de projectrol (bijvoorbeeld toegang tot projectmap).

Voorbeeld van role engineering

In een eenvoudig voorbeeld zijn voor de medewerkers van kostenplaats 192538 de autorisaties weergegeven die zij hebben in het cliëntacceptatiesysteem (CAS), een databasemanagementsysteem (DBMS), in Unix en in Active Directory op Windows (WinAD). Als we de matrix bestuderen dan vallen enkele zaken op:

- Eén van de acceptanten – A. Willems – heeft een autorisatie Ccgrp4 in WinAD, die de collega-acceptant niet heeft.
- Eén van de baliemedewerkers – E. Ronner – heeft niet de autorisatie Ccgrp 2 in WinAD, terwijl de collega-baliemedewerkers deze autorisatie wel hebben.
- Eén van de casemanagers – E. Coops – heeft in DBMS03 de autorisatie Tbl01Delete, terwijl de collega's deze niet hebben.
- Casemanager G. Dam is in WinAD niet gekoppeld aan Reggrp, terwijl de collega-casemanagers deze autorisatie wel hebben; ook valt op dat G. Dam als enige casemanager geautoriseerd is voor Access.
- R. Ginneken heeft als enige niet de autorisatie 192538_mappen in WinAD.

Na bestudering van de matrix leggen we deze aan de lijnmanager voor; dan blijkt dat de afwijkende autorisaties van Willems en Coops inderdaad een kwestie van vervuiling waren en dat het ontbreken van autorisatie Reggrp bij G. Dam, van Ccgrp2 bij E. Ronner en van 192538_mappen bij R. Ginneken onterecht is. Verder licht de lijnmanager toe dat voor het gebruik van MS-Access licentiekosten moeten worden betaald en dat daarom alleen hijzelf en G. Dam hiervoor geautoriseerd zijn. Daarnaast valt het de lijnmanager op dat hij nog steeds een oude (overbodige) autorisatie heeft: Client_Acceptant in CAS.

Na het raadplegen van diverse beheerders noteren we aanvullend de volgende gegevens over de autorisaties:

- Om CAS te kunnen gebruiken is – behalve een applicatierol Client_Acceptant of Client_onderhoud – ook altijd de autorisatie CASgrp op Unix02 nodig.
- Autorisaties Tbl01View in DBMS03 en Repgrp in WinAD zijn altijd beide nodig om cliëntgegevens te kunnen rapporteren.

Kostenplaats 192538		Medewerker	Dollé, C.	Willems, A.	Ginneken, R.	Jansen, J.	Ronner, E.	Coops, E.	Dam, G.	Pieper, P.	Son, R. van
Systeem	Autorisatie	Rol	Acceptant	Acceptant	Baliemedew.	Baliemedew.	Baliemedew.	Casemanager	Casemanager	Casemanager	Teamleider
CAS	Client_Acceptant		X	X							X
CAS	Client_onderhoud							X	X	X	
DBMS03	Tbl01View							X	X	X	X
DBMS03	Tbl01Delete							X			X
Unix02	CASgrp		X	X				X	X	X	
WinAD	192538_mappen		X	X		X	X	X	X	X	X
WinAD	Access								X		X
WinAD	Allusers		X	X	X	X	X	X	X	X	X
WinAD	Ccgrp				X	X	X	X	X	X	X
WinAD	Ccgrp2				X	X		X	X	X	X
WinAD	Ccgrp4			X	X	X	X	X	X	X	X
WinAD	Reggrp				X	X	X	X		X	
WinAD	Repgrp							X	X	X	X

Figuur 6. Matrix met bestaande autorisaties.

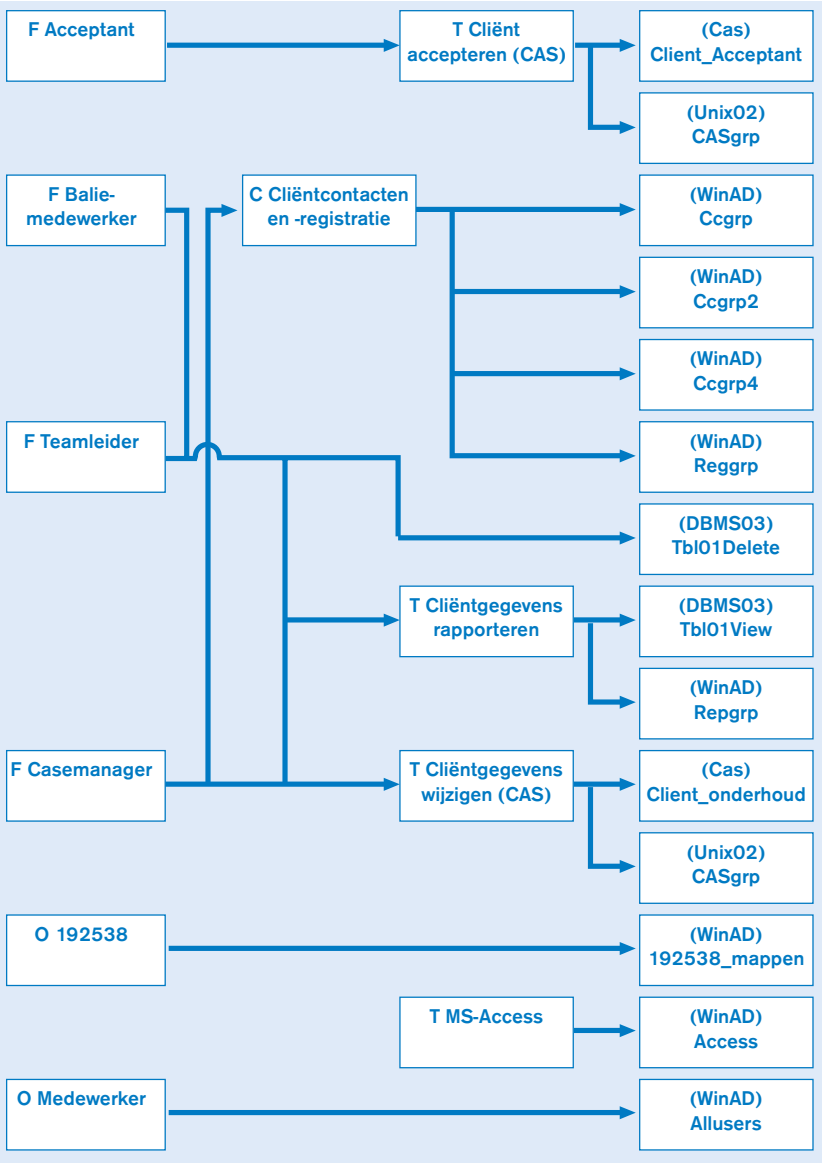
Alle autorisaties worden als permissies in het RBAC-tool opgenomen, en op basis van de verkregen informatie gegroepeerd in rollen, zoals in figuur 7. Hierbij wordt als naamgevingsconventie gehanteerd dat de naam van functie-/procesgerelateerde rollen begint met een F, van connectorrollen met een C, van organisatierollen met een O, en van taakgebaseerde subrollen met een T.

Na het inrichten van de rollen en permissies kunnen de gebruikers worden gekoppeld aan de gedefinieerde rollen: elke medewerker wordt gekoppeld aan de betreffende functie-/procesgerelateerde rol, en tevens aan de beide organisatorische rollen.

Afhankelijk van de tooling kan ook (slim!) gebruik worden gemaakt van '*Rule* Based Access Control': waar nu in figuur 7 gekozen is voor twee organisatorische rollen, kan ook op basis van een 'rule' – zonder tussenkomst van een rol – worden gerealiseerd dat de gebruikers de organisatiegerelateerde autorisaties 192538_mappen en Allusers in WinAD krijgen. Een andere oplossing voor het elimineren van de rol 'O Medewerker' is om gebruik te maken van rolhiërarchieën in het tool, waardoor deze rol (impliciet) geërfd wordt via de rol 'O 192538'.

Zoals er vele wegen naar Rome leiden, zo geldt ook hier dat het model op veel verschillende manieren kan worden ingericht. De oplossing in figuur 7 lijkt te voldoen aan de gestelde eisen van transparantie en onderhoudbaarheid. Enkele voorbeelden:

- Als voor 'cliëntcontacten en -registratie' een nieuwe permissie haar intrede doet, bijvoorbeeld Ccreg01 op Unix02, dan hoeft deze slechts aan één rol te worden gekoppeld: aan de connectorrol 'C Cliëntcontacten en -registratie', in plaats van afzonderlijk aan de drie rollen 'F Baliemedewerker', 'F Teamleider' en 'F Casemanager'.



Figuur 7. Permissies ondergebracht in het RBAC-model.

- Als de baliewerkzaamheden worden overgeheveld naar een andere kostenplaats, dan hoeft bij de betrokken medewerkers enkel de koppeling aan 'O 192538' te worden vervangen door de organisatorische rol van de nieuwe afdeling.
- Als acceptanten ook moeten kunnen gaan rapporteren over cliëntgegevens, dan volstaat het koppelen van de taakrol 'T Cliëntgegevens rapporteren' aan hun rol 'F Acceptant', in plaats van het afzonderlijk koppelen van de permissies 'Tbl01View' en 'Repgrp' aan hun rol.

Dit zijn slechts enkele voorbeelden, die op het eerste gezicht niet zo heel indrukwekkend zijn, maar hopelijk wel duidelijk maken hoe vooral in complexere, omvangrijkere situaties de transparantie en onderhoudbaarheid gebaat zijn bij het 'slim' inrichten van het RBAC-model.

Conclusie

Als de organisatie geen of onvoldoende beeld heeft van de benodigde autorisaties per rol, dan vormt role engineering een omvangrijke klus in het RBAC-implementatietraject. Role mining-tools kunnen hierbij ondersteunen, maar meestal zal pas uit gesprekken met onder meer autorisatiebeheerders, functioneel beheerders, technisch beheerders en database administrators blijken wat de werkelijke redenen zijn voor de autorisaties. Gegeven de variatie in platforms, middleware en applicaties, en de raakvlakken met AO/IC-aspecten, moeten hiervoor 'zware' RBAC-projectmedewerkers worden ingezet, die zowel de taal van de lijnmanager spreken, als de talen van de verschillende beheerders. Als aan deze voorwaarde wordt voldaan, en uit de gesprekken met beheerders het vereiste inzicht is verkregen, dan is het mogelijk om tot een zodanige invulling van het RBAC-model te komen dat de transparantie en onderhoudbaarheid van het autorisatiemodel voldoen aan de gestelde interne en externe eisen.

Literatuur

[Ferr03] David F. Ferraiolo, D. Richard Kuhn and Ramaswamy Chandramouli, *Role-Based Access Control*, Artech House, 2003.

[Hofm05] Ir. A. Hofman, *Rolgebaseerd autoriseren onder architectuur*, Informatiebeveiliging 2005/1.

[Koor04] Drs. ing. R.F. Koorn RE en ing. J.A.M. Hermans RE, *Identity Management: hoe (on)toereikend is het nu en hoe kan het beter?*, Compact 2004/2.

[Kuhl03] Martin Kuhlmann, Gerhard Schimpf and Dalia Shohat, *Role mining – Revealing Business Roles for Security Administration using Data Mining Technology,* ACM Press 2003.

[Mien03a] Ing. P. Mienes RE en B. Bokhorst RE RA, *De (on)beheersbaarheid van toegangsbeveiliging,* Compact 2003/1.

[Mien03b] Ing. P. Mienes RE en B. Bokhorst RE RA, *RBAC: gewoon doen*, Informatiebeveiliging 2003/2 en 2003/3.