Vraag 1 AD replicatie

1.a Wat is de bedoeling van replicatie?

AD replicatie verzorgt de verspreiding van de directory gegevens over meerdere domeincontrollers. Alle domeincontrollers in een domein nemen deel aan de replicatie, en bevatten een volledige kopie van alle directory gegevens van het eigen domein. Alle domeincontrollers van hetzelfde forest beschikken over hetzelfde directory schema en dezelfde configuratiegegevens. Meerdere domeincontrollers zijn nuttig om de belasting te spreiden en de fouttolerantie/beschikbaarheid te verhogen.

1.b Hoe wordt dit in Windows Server geraliseerd? (onder meer ten opzichte van NT 4.0) Bespreek de verschillende technische kenmerken en concepten van Windows Server replicatie.

Windows NT4.0 werd replicatie georganiseerd met een *master-slave model*. In dit model beschikt enkel de PDC over een wijzigbare kopie van de gegevens. De andere *backup domain controllers* waren alleen-lezen.

Windows Server gebruikt *multi-master replicatie*. Elke domeincontroller bevat een wijzigbare kopie van de gegevens, tenzij hij RODC is. Elke verandering op een willekeurige domeincontroller wordt verspreid naar alle andere domeincontrollers. Alle domeincontrollers zijn dus equivalent. Dit biedt verhoogde fouttolerantie. Als een domeincontroller uitvalt, kan de replicatie worden voortgezet tussen de overgebleven domeincontrollers.

Het is belangrijk om het replicatieverkeer zo veel mogelijk te beperken. Daarom worden enkel gewijzigde gegevens gerepliceerd. Verder moet er ook worden gelet dat een zelfde wijziging niet meermaals naar dezelfde domeincontroller wordt gerepliceerd. Hiervoor wordt het *Update Sequence Number* gebruikt, dat wordt verhoogd telkens een object wordt aangepast. Het USN samen met de GUID van de domeincontroller vormt de *Up-to-dateness Vector*. Elke domeincontroller meldt zijn UTD vector aan alle andere domeincontrollers telkens er een wijziging aan een object gebeurt. Elke domeincontroller houd ook een UTD vector tabel bij, die de meest recente UTD vector van alle andere domein controllers waarvan hij wijzigingen heeft verwerkt, bevat. Door de UTD vectortabel van twee domeincontrollers te vergelijken, kan worden nagegaan of deze elkaars wijzigingen hebben uitgewisseld.

De metadata van objecten houden voor elk kenmerk een *property version number* bij, samen met de UTD vector van de domeincontroller die de wijziging heeft uitgevoerd. Aan de hand van deze kenmerken kan een domeincontroller exact bepalen welke gegevens bij moet opsturen.

Potentiële conflicten worden opgelost door rekening te houden met de recentste wijziging en grootste GUID van de domeincontroller van oorsprong. De wijziging die deze kenmerken heeft wordt geaccepteerd, de andere wordt verworpen.

Om te vermijden dat verwijderde objecten opnieuw worden gecreëerd door replicatie, worden deze gedurende een bepaalde tijdspanne als *tombstones* in een verboren container geplaatst. Na afloop van de *tombStoneLifteTime* worden de objecten verwijderd.

Een tweede aspect is *store-and-forward* replicatie. In dit principe worden de wijzigingen slechts uitgewisseld met bepaalde andere domein controllers. Elk van deze domeincontrollers wisselen de gegevens op hun beurt dan uit met andere domeincontrollers, enzovoort. Deze *replicatietopologie* wordt automatisch opgesteld koor de *Knowledge Consistency Checker* van AD. De configuratie gebeurt met behulp van verbindingsobjecten, die in enkele richting werken. Replicatierelaties zijn dus niet noodzakelijk wederzijds. Dit is zeker niet het geval indien er RODC's in het spel zijn. De ontstane topologie voldoet aan enkele voorwaarden:

- De topologie is optimaal.
- Voor forestgegevens en domeingegevens worden aparte topologieën geconstrueerd.
- Elke controller is met maximaal drie andere controllers verbonden
- Er zijn minstens twee paden die elke domeincontroller bevatten
- Het grootste aantal hops tussen twee willekeurige domeincontrollers is 3

De topologie wordt periodiek gecontroleerd, en aangepast indien nodig. Standaard gebeurt dit elk kwartier.

Het replicatie mechanisme is *pull* based. Controllers vragen zelf om gewijzigde gegevens. Partners brengen mekaar enkel op de hoogte van de wijzigingen. De wijzigingen worden gebundeld en periodiek verstuurd, dit heet *propagation damping*.

Windows Server replicatie is *multithreaded*. Replicatie kan met verschillende partners tegelijk gebeuren.

Een derde aspect is de *kleinste replicatie-entiteit*. In NT4.0 is de kleinste entiteit die in zijn geheel kan worden gerepliceerd een object. In Windows Server is dit een individueel kenmerk (2000) of een atomaire waarde van een multi-valued kenmerk (2003+)

1.c Welke toestellen repliceren onderling in een forest? Welke gegevens worden hierbij uitgewisseld?

Tussen een domeincontroller met *global catalog server* van het ene domein en een domeincontroller van het andere domein worden het schema en de configuratiegegevens gerepliceerd. De domeincontroller zal ook een subset van zijn domeingegevens naar de global catalog server repliceren. Tussen de global catalog servers van een forest worden subsets van alle domeingegevens alle domeinen gerepliceerd.

1.d Welke impact hebben sites op het replicatieverkeer? Je hoeft hierbij het site op zichzelf niet verder te behandelen.

Binnen de site verloopt de replicatie zoals normaal. Inter-site replicatie vertoont implementatie verschillen intra-site replicatie. De bedoeling is om de juiste balans te vinden tussen de actualiteit van de gegevens en de bandbreedtebeperkingen.

- Adverteren van wijzigen in UTD vector tabellen worden achterwege gelaten, maar kan worden ingeschakeld. Enkel het polling mechanisme blijft.
- Replicatieverkeer tussen sites wordt gecomprimeerd
- De KCC software kan worden beïnvloed met *sitekoppelingen*. De KCC legt enkel een replicatieverbinding tussen sites als er een koppeling tussen bestaat. Sites moeten manueel verbonden worden met sitekoppelingen, anders blijven ze geïsoleerd.

Vraag 2 AD sites

2.a Welke rol vervullen sites? Welke AD aspecten worden er door beïnvloed en hoe?

Sites zijn een weerspiegeling van de fysieke locaties van bepaalde toestellen. Deze fysische indeling staat los van de logische indeling in forest, trees, domeinen, of OU's. Een site is typisch een verzameling van subnetwerken die onderling met hoge bandbreedte verbonden zijn. Sites zijn dan meestal met lagere bandbreedten verbonden. Beperken van het verkeer is meteen ook het voornaamste aspect van AD sites.

De replicatie tussen domeincontrollers van verschillende sites wordt zoveel mogelijk beperkt, en moet expliciet worden geconfigureerd.

De KCC software duidt per site een *inter site topology generator* aan. Enkele deze domeincontroller heeft de toestemming om informatie in sitekoppelingen te gebruiken voor replicatieverkeer. Er wordt dan hoogstens een verbindingsobject tussen de controllers van de verschillende sites aangemaakt. Controllers die van dit verbindingsobject gebruik maken, worden bruggenhoofden genoemd.

2.b Welke relaties bestaan er tussen sites, domeinen, domein controllers en global catalogs?

Een domein kan meerdere sites overspannen. Tegelijk kunnen op eenzelfde site ook meerdere domeinen aanwezig zijn.

Een toestel zal bij voorkeur steeds een domeincontroller van zijn eigen site contacteren. Ook is het aangeraden om minstens een domeincontroller met global catalog server per site te plaatsen. Dit opnieuw om het verkeer over de grenzen van de site zoveel mogelijk te beperken.

2.c Hoe wordt bepaald tot welke site computers behoren?

Een site is een verzameling van subnetwerken die onderling met hoge bandbreedte kunnen communiceren. Als een toestel een interface in een van die subnetwerken heeft, dan behoort hij tot de site waartoe zijn subnetwerk behoort. Als een toestel interfaces in meerdere subnetwerken heeft (bijvoorbeeld routers) kan hij tot slechts een site behoren. De fysieke locatie van het toestel is dan een goede leidraad.

Werkposten worden dynamisch aan sites gekoppeld. Telkens die IP software opstart, bepaalt het netwerkadres de site waarin hij zich bevindt.

De site van een domeincontroller wordt bepaald door de *Servers* container waarin het server object van de domeincontroller zich bevindt. Elke site heeft deze container. De initiële bepaling gebeurt met dezelfde regels als voor werkposten.

Toestellen met een adres dat niet aan een site kan worden gerelateerd, komt in de *Default-Firs-Site* site terecht.

2.d Bespreek de diverse noodzakelijke instellingen om de verschillende aspecten van sites te configureren,en vermeld hierbij telkens waarom ze noodzakelijk zijn en waar ze opgeslagen worden.

De meeste configuratie van sites wordt in AD zelf bijgehouden, meerbepaald in de *Sites* container van de *configuratiegegevens*. Sommige instellingen van het replicatiemechanisme worden in het register van elke individuele domeincontroller bijgehouden. Deze instellingen staan in de *Parameters subtak* van de *NTDS service.*

Elementaire configuratie gebeurt met de *Active Directory Sites and Services* snap-in, in *dssite.msc*.

Creatie van sites en toevoegen van lidservers aan sites kan in *dssite.msc*. Ook sitekoppelingen worden hier aangemaakt, deze bevinden zich in de *Inter Site Transport Link* container van de snap-in. Door op deze container rechts te klikken, *IP* aan te klikken en vervolgens *New Site Link* aan te klikken kan een nieuwe link worden aangemaakt. Let erop dat sitekoppeling in tegenstelling tot verbindingsobjecten wél reflexief zijn. Ze moeten niet in beide richtingen worden aangemaakt. Sitekoppelingen zijn niet transitief. Bijkomende instellingen zijn:

- Het *protocol* dat voor het replicatieverkeer wordt gebruikt. Mogelijkheden zijn *RPC* en *SMTP*. RPC is betrouwbaarder maar veroorzaakt meer belasting. SMTP vergt minder resources.
- Het *synchronisatie schema* bepaalt de tijdstippen waarop de synchronisatie gebeurt.
- De *bandbreedte* en de *kost* van de verbinding kunnen worden ingesteld. De goedkoopste verbinding die beschikbaar wordt gebruikt.
- Het *interval* tussen verschillende polls.

Voor meer gedetailleerde configuratie moet het object van de sitekoppeling rechtstreeks worden aangepast. Bijvoorbeeld met *ADSIEdit.msc*.

Alle verbindingsobjecten, zowel inter- als intra-site, zijn opgenomen in de *NTDS Settings* container van de *dssite.msc* snap-in. De manuele creatie (om een gerichte replicatietopologie te verkrijgen) van verbindingsobjecten is mogelijk in deze container. Meestal is het beter om de KCC deels of volledig zijn gang te laten gaan.

Subnetten vormen de basis voor de indeling in sites. Elke site geassocieerd met minstens één subnet, en minstens één sitekoppeling. Werkposten worden toegewezen aan sites op basis van hun adres. Subnetten kunnen worden gecreëerd in de container *subnets* van *dssite.msc.*

Het bruggenhoofd, die als enige domeincontroller gegevens mag repliceren over de sitegrenzen heen, kan expliciet worden geconfigureerd in de *properties tabpagina* van de server.

De *site covering* van een domeincontroller bepaalt welke sites hij kan bedienen, in het geval dat alle domeincontrollers in een bepaalde site zijn uitgevallen.

Sites krijgen best een eigen *global catalog* toegewezen. Een domeincontroller tot global catalog server promoveren kan ook in *dssite.msc*, meer bepaald in de *general tabpagina* van de *properties* van de *NTDS* settings van de overeenkomstige server.

In de properties van de *NTDS Site Settings* kan de *Inter-Site Topology Generator* voor de site worden gekozen. Ook *Universal Group Membership Caching* kan hier worden ingeschakeld. Dankzij deze optie is het in 2003+ domeinen mogelijk om de inlogprocedure te voltooien zonder een global catalog server te moeten contacteren. De cache wordt in AD zelf opgeslagen, maar wordt niet gerepliceerd.

Vraag 3 Machtigingen op bestandstoegangen

3.a Welke rol spelen machtigingen bij de beveiliging van bronnen? Geef een gedetailleerd algemeen overzicht van het mechanisme van machtigingen.

Machtigingen op een bepaalde bron bepalen wie toegang heeft tot de bron, en wat die er mee kan doen.

Elk object in AD, elk object van een NTFS volume, elke registersleutel, elk proces, en elke service heeft een *security descriptor*. Deze bevat:

- Een *machtigingsset* of *Discretionary Access Control List*. (ACL)
- Een *System Access Control List* doe definieert welke acties van de gebruiker gelogd worden
- De *SID* van de eigenaar van het object. Makers van objecten zijn standaard eigenaar, maar de eigendom kan worden overgedragen. De eigenaar of beheerder is verantwoordelijk voor het instellen van de ACL
- Wegens POSIX compatibiliteit: de *primary group* van de maker.

De ACL is een verzameling van machtigingen die bepaalt welke gebruiker of welke groep welke toegangsrechten heeft voor het object. De specifieke machtigingen die kunnen worden toegekend is afhankelijk van het object waarop ze worden toegepast. De ACL bestaat uit *Access Control Entries* of machtigingsvermeldingen. Een ACE kan zowel een machtiging toekennen als ontzeggen. Machtigingen worden best zo veel mogelijk op groepen toegepast om beheer te vereenvoudigen. ACE's worden in *cannonieke volgorde* verwerkt. Eerst komen de ACE's die machtigingen ontzeggen aan de beurt, daarna degene die toekennen. Ontzeggen is steeds het sterkste kenmerkn, een toekenning van een machtiging wordt genegeerd als die eerder ontzegt werd. Machtigingen zijn *cummulatief*. De machtigingen van een groep worden op zijn leden toegepast, tenzij die machtiging elders (op de specifieke gebruiker of een andere groep) ontzegd werden. Machtigingen worden impliciet gewijgerd. Afwijken van de cannonieke volgorde kan enkel door programmatisch de volgorde van de ACE's in het object te wijzigen.

Er zijn twee soorten machtigingen:

- *Expliciete machtigingen*: rechtstreeks aan het object gekoppeld.
- *Overgenomen machtigingen*: machtigingen overgenomen van de container waartoe het object behoort. Dit vereenvoudigt het beheer van machtigingen sterk.

Expliciete machtigingen krijgen altijd voorrang op impliciete.

Het ontbreken van een ACL is een ernstig risico. Immers objecten zonder ACL zijn voor iedereen toegankelijk, terwijl een lege ACL ervoor zorgt dat toegang impliciet wordt geweigerd.

3.b Bespreek hoe het mechanisme voor machtigingen specifiek wordt toegepast op bestandstoegang. Geef de verschillende soorten machtigingen, hun onderlinge relaties en hoe deze kunnen geanalyseerd worden. Toon hierbij aan dat je zelf met deze configuratietools geëxperimenteerd hebt.

NTFS machtigingen bestaan in twee niveaus. *Atomaire* machtigingen vormen de bouwstenen voor *moleculaire* machtigingen. Moleculaire machtigingen zijn veelgebruikte combinaties van atomaire machtigingen.

De 13 atomaire machtigigingen zijn:

- *Traverse Folder/Execute File:* Traverse Folder geldt voor mappen en laat de gebruiker toe om hulpbestanden die zich meer dan een niveau lager bevinden waartoe de gebruiker toegang heeft, maar niet de tussenliggende mappen toch te gebruiken.
- *List Folder/Read Data:* Inhoud van een bestand of map tonen.
- *Read Attributes:* Elementaire bestandskenmerken weergeven.
- *Read Extended Attributes:* Programma-afhankelijke uitgebreide bestandskenmerken weergeven.
- *Create Files/Write Data:* Nieuwe bestanden in een map aanmaken/Data van een bestaand bestand overschrijven, maar niet aan een bestaand bestand toevoegen.
- *Create Folder/Append Data*: Nieuwe mappen creëren en data toevoegen aan een bestand.
- *Write Attributes*: Elementaire bestandskenmerken wijzigen.
- *Write Extended Attributes:* Uitgebreide bestandskenmerken wijzigen.
- *Delete Subfolders and Files:*
- *Delete:* Het object zelf verwijderen.
- *Read Permissions:* NTFS machtigingen weergeven.
- *Change Permissions:* NTFS machtigingen wijzigen.
- *Take Ownership*: eigenaarschap van een bestand of map overnemen. De eigenaar staat buiten alle machtigingen en kan de machtigingen van het object wijzigen.

De 6 moleculaire machtigingen:

- *Full Control*: Alle atomaire machtigingen
- *Read*: Inhoud, machtigingen en kenmerken van een object bekijken. Submappen waar geen toegang tot is blijven zichtbaar.

- *Read&Execute*: Identiek aan de *Read* machtiging, aangevuld met de atomaire *Traverse Folder/Execute File* machtiging.

- *Write*: Nieuwe bestanden en submappen aanmaken. Om bestaande bestanden te kunnen wijzigen is de aanvullende *Read* moleculaire machtiging nodig

- *Modify*: *Read&Execute + Write + Delete* (atom.)

- *List Folder Contents*: enkel voor mappen, laat toe om inhoud van mappen te bekijken ongeacht het bezit van *Read* toegang.

De ACL van een object bevindt zich in de *security* tabpagina van het *properties* venster. In NT6+ moet eerst op *edit* worden geklikt om de ACL te wijzigen. In het *name* paneel staan de gebruikers en groepen waarvoor de machtigingen gelden. De moleculaire machtigingen voor de geselecteerde gebruiker of groep verschijnen in het *Permissions* paneel. Grijze selectievakjes duiden erop dat de machtiging werdt overgenomen van een bovenliggende map.

De knop *Advanced* geeft toegang tot een volgend niveau van machtigingen. Er zijn 4 tabpagina's.

De *Permissions* tabpagina toont een overzicht van ACE's in de volgorde zoals de door de SRM worden verwerkt. De kolom *Inherited From* geeft aan van waar het attribuut werd overgeërfd. De knop *Edit* opent het venster in *edit mode*. Door een ACE te selecteren in *edit mode* komt een lijst van atomaire machtigingen tevoorschijn. Speciale combinaties van atomaire machtigingen die niet tot een combinatie van moleculaire machtigingen te herleiden zijn kunnen hier worden ingesteld. In het *properties* venster wordt dit weergegeven als *Special Permissions*. De *Permissions Tabpagina* in edit mode laat drie aanvullende mogelijkheden toe:

- Het selectievakje *Allow Inheritable Permissions* geeft aan of er machtigingen worden overgeërfd van de bovenliggende map. Bij het uitschakelen wordt gevraagd of bestaande overgeërfde permissies van de bovenliggende map moeten worden gekopieerd of verwijderd.

- Het selectievakje *Replace all existing inheritable permissions* zorgt ervoor dat, indien ingeschakeld, alle machtigingen op onderliggende objecten expliciet worden toegepast. Hierbij worden alle machtigingen van de onderliggende objecten vervangen. Reeds gedefinieerde machtigingen gaan verloren. Deze actie kan niet ongedaan worden gemaakt.

- Met de knoppen *Add/Remove/Edit* kunnen ACE's worden toegevoegd, verwijderd of de atomaire machtigingen worden ingesteld. De keuzelijst *apply onto* kan overerving selectief laten gelden.

De *Auditing* tabpagina wordt gebruikt om de SACL van het object in te stellen.

De *Owner* tabpagina laat toe om de eigenaar van het object te veranderen. Dit kan enkel gebeuren door gebruikers met de *Take Ownership* machtiging op het object, die enkel kan worden toegekend door beheerders, eigenaars of gebruikers met *Full Control*. Overdragingen van eigendom moeten wederzijds aanvaard worden.

De *Effective Permissions* tabpagina toont een overzicht van de uiteindelijke effectieve machtigingen zoals ze zouden worden toegekend door de *Security Reference Monitor*. Dit kan handig zijn voor probleem diagnose.

In de *Command Prompt* kunnen machtigingen worden bekeken met *showacls* of *perms.* Wijzigingen aanbrengen kan met *icacls*, *xcacls* of *subinacl*. De eerste laat toe om een lijst van permissies te bewaren in een bestand, om dan later met de optie */restore* te herstellen.

3.c Wat gebeurt er met de machtigingen bij het kopiëren of verplaatsen van een bestand?

De gebruiker die de actie onderneemt wordt eigenaar van de bestanden wanneer ze bij de bestemming aankomen. Waneer de bestemming een container is op een andere NTFS volume, of de bestanden met standaard tools werden *gekopieerd*, vervallen de expliciete machtigingen. De machtigingen van de doelcontainer worden overgenomen door het object zelf en al zijn onderliggende objecten. Objecten die naar een niet NTFS volume worden gekopiëerd, verliezen alle machtigingen.

Om de machtigingen tijdens een *kopieeropdracht* te behouden moeten speciale tools zoals *robocopy* en *scopy* worden gebruikt.

Wanneer bestanden of mappen worden verplaatst naar een container binnen hetzelfde volume, worden de expliciete machtigingen behouden, en worden de machtigingen van de container overgenomen.

3.d Welke andere objecten hebben machtigingen?

Elk object in AD, elk object van een NTFS volume, elke registersleutel, elk proces, en elke service heeft een *security descriptor*, en dus de bijbehorende machtigingen.

Vraag 4 Configuratie van domeinaccounts

(Handgeschreven)

Vraag 5 Gebruikersgroepen

5.a Bespreek in detail het verschil tussen de verschillende soorten veiligheidsgroepen. Behandel hierbij vooral de mogelijkheden en beperkingen. Welke zijn bijvoorbeeld (beperk je niet tot deze aspecten!) de onderlinge relaties en de regels voor het nesten van de diverse soorten groepen? Stel deze zoveel mogelijk schematisch voor.

	Zichtbaarheid	Geldigheid	Aanwezig op:	Bevat
Domein lokale groep	Alle lidcomputers (workstations en servers) van het eigen domein.	Eigen Domein	DC van eigen domein.	Gebruikers en groepen uit elk domein van het forest of een ander trusted domein. Lokale groepen van het eigen domein.
Globale groep	Elk domein van het forest of een ander trusting domein.	Volledige forest.	DC en eigen domein en GC (enkel naam op GC, geen leden)	Gebruikers en globale groepen van het eigen domein.
Universele groep	Alle domeinen van het forest.	Alle domeinen van het forest.	GC van het forest.	Gebruikers en groepen met globale scope van elk domein uit het forest.

5.b Hoe en waarom worden deze soorten groepen in de praktijk best (niet) gebruikt? Van welke omstandigheden is dit afhankelijk? Illustreer aan de hand van voorbeelden.

Lokale groepen worden meestal gebruikt om rechten en machtigingen toe te kennen binnen hetzelfde domein, en bevatten eerder andere groepen dan gebruikers. Men creëert een lokale groep, die men toegang geeft tot de bron en voegt andere groepen aan deze groep toe, om te toegang te verlenen. Lokale groepen zijn interessant wanneer de groep enkel zichtbaar mag zijn binnen een domein.

Globale groepen worden gebruikt om gebruikers van een domein te groeperen. Globale groepen kunnen rechten worden toegewezen in een ander domein, bijvoorbeeld door ze toe te voegen aan een lokale groep van het domein waartoe de bron behoort, die toegang heeft tot de bron. Globale groepen zijn minder interessant om aan een bron te koppelen, omdat ze enkel

gebruikers en andere globale groepen kunnen bevatten. Lokale groepen kunnen zowel gebruikers als elk soort andere groepen bevatten.

Universele groepen kunnen voor dezelfde doeleinden worden gebruikt al domein lokale groepen, maar zijn dan onmiddellijk zichtbaar in het gehele forest. Dit heeft dus het bijkomend voordeel dat ze niet in elk domein opnieuw moeten gedefinieerd worden. Een mogelijke strategie laat lokale en universele groepen achterwege, en gebruikt enkel universele groepen. Dit heeft echter als risico dat de GC onnodig groot wordt, omdat elke universele groep met al zijn leden in de GC wordt opgenomen. Indien er nog frequente wijzigingen aan het lidmaatschap gebeuren, veroorzaakt dit bijkomend replicatieverkeer. In 2003+ is dit probleem beperkter omdat niet de gehele lijst van leden moet worden gerepliceerd, maar enkel de individule elementen van het multivalued attribuut. Bovendien is het contacteren van een GC nodig voor het inloggen van een gebruiker, om het lidmaatschap van een universele groep na te gaan.

5.c Waar en hoe wordt het (volledige) lidmaatschap van een object tot een groep bijgehouden? Op welke diverse manieren kan men dit lidmaatschap configureren? Door wie wordt het lidmaatschap bij voorkeur ingesteld?

De groepen waartoe een object behoort worden opgeslagen in het *memberOf* attribuut. Omdat het een *back-link* attribuut is, kan het niet rechtstreeks worden gewijzigd. Objecten programmatisch aan groepen toe te voegen, kan door het *member* attribuut van het groepsobject aan te vullen. Dit is het corresponderende *forward-link* attribuut van *memberOf*.

Om met de grafische interface een gebruiker of groep aan een andere groep toe te voegen, kan het *Member Of* tabblad van het *Properties* venster van de gebruiker of groep worden gebruikt.

5.d Welke conversieregels bestaan er tussen verschillende soort groepen?

Lokale groepen kunnen naar de universele scope worden geconverteerd, op voorwaarde dat ze geen andere lokale groepen bevatten.

Globale groepen kunnen naar de universele scope worden geconverteerd, zolang de groep geen lid is van een andere groep met globale scope.

Universele groepen kunnen meestal niet worden omgezet naar anders scopes.

Veiligheidsgroepen kunnen worden omgezet naar distributiegroepen, en omgekeerd.

5.e Welke groepen worden in de praktijk rechten toegekend? Bespreek de bijzonderheden van dergelijke groepen en vermeld er een aantal voorbeelden van (telkens met hun bedoeling.)

Builtin lokale groepen krijgen een combinatie van rechten. Rechten voor omlijnde beheerstaken uit te voeren, kunnen zo snel worden toegekend, zonder dat ze vanaf nul moeten worden opgebouwd.

- *Backup Operators*: Hebben de bevoegdheid om bestanden en mappen te backup-uppen of te herstellen, ongeacht de rechten om te schrijven of te wijzigen. De rechten hebben voorrang op de machtigingen, tijdens het maken backups

- *Account Operators*: Maken en beheren gebruikersaccounts, kunnen computers aan het domein toevoegen. Kunnen onder andere de builtin lokale groepen en *Domain Admins* groep niet wijzigen. Kunnen inloggen op servers, ze herstarten en afsluiten.

- *Server Operators*: Kunnen systemen vanop afstand uitschakelen, systeemtijd veranderen, schijven formatteren en mappen delen.

- *Administrators*: Hebben bijna elk recht. Ze kunnen wel worden buitengesloten van bepaalde bestanden en mappen.

- *Users*: kunnen programma's gebruiken maar ze niet installeren. Users hebben het recht lokale groepen aan te maken en ze te beheren.

Impliciete groepen of *Speciale Identiteiten* beschrijven een aantal gebruikers. Het lidmaatschap is niet wijzigbaar en kan niet worden weergegeven. De lidmaatschap van deze groep verandert zonodig. Het zijn geen echte groepen.

- *Interactive*: groepeert iedereen die een toestel lokaal gebruikt, *Network* alle gebruikers die via het netwerk met een bron van het toestel verbonden zijn. Beide groepen vormen samen de groep *Everyone*. *Authenticated Users* is een veilger alternatief voor *Everyone*.

- *System* vertegenwoordigt het besturingssysteem. Deze gebruiker heeft alle rechten.

- *Creator Owner* krijgt alle machtigingen toegekend en heeft volledige toegang tot het object.

- *Owner Rights* wordt de machtiging *Change Permissions* ontzegd om te voorkomen dat de *Creator Owner* de security descriptor kan instellen.

Vraag 6 Gebruikersprofielen

6.a Wat is het nut van gebruikersprofielen? Bespreek zowel vanuit het standpunt van gebruikers, als van beheerders, en bespreek hierbij de voordelen en de nadelen.

De desktop instellingen van een gebruiker worden in NT4+ automatische bewaard in een gebruikersprofiel. Het een aangepaste desktop omgeving met individuele weergave instellingen, netwerkverbindingen, printerverbindingen en andere instellingen.

Vanuit gebruikersstandpunt biedt het gebruik van *lokale* profielen een drietal voordelen. Ten eerste beschikt elke gebruiker over een persoonlijke desktop omgeving. Instellingen die in de eigen omgeving worden gemaakt, hebben geen invloed voor de andere gebruikers. Desktop instellingen die tijdens een vorige sessie actief waren, worden bij een volgende sessie hersteld.

Lokale profielen zijn vanuit beheersstandpunt veruit de gemakkelijkste manier om de gebruikersinterface te configureren. De configuratie kan ook gebeuren voor specifieke gebruikers.

Het opslaan van de profielen op een centrale server biedt bijkomende voordelen vanuit beide standpunten. In dit geval spreekt men van *roaming profiles*. Zo zijn de instellingen van de gebruiker op elke willekeurige machine van het netwerk beschikbaar. Een nadeel is een langere inlog- en uitlogprocedure: de profielen moeten immers van en naar de server worden gekopieerd bij aan- en afmelden.

Vanuit beheersstandpunt biedt het gebruik van *roaming profiles* de voordelen van een verhoogde fouttolerantie en een betere centrale beheersbaarheid. Nadelig is echter de bijkomende netwerkbelasting die het synchronisatie verkeer met zich meebrengen. Profielen worden op werkposten ook lokaal gecachet. Dit houdt een veiligheidsrisico in. Beheerders moeten zich ook bekommeren over de compatibiliteit van de profielen tussen verschillende versies van Windows. (NT6- en NT6+).

Zie ook *mandatory roaming profiles* en *super mandatory roaming profiles*.

6.b Geef de verschillende soorten profielen Hoe worden deze ingesteld, en waar worden ze bij voorkeur opgeslagen?

Gebruikersprofielen bestaan in twee smaken. *Lokale profielen* en *roaming profiles*.

Lokale profielen bestaan enkel op de schijf van het toestel. Ze worden op geslagen in de *Users* (NT6+) of *Documents and Settings* (NT5) submap van de systeempartitie. Profielmappen krijgen eventueel de domeinnaam als suffix om onderscheid te maken met reeds bestaande profielen.

Een *roaming profile* verschilt van een lokaal profiel in het feit dat ze steeds worden gesynchroniseerd met de kopie op de server. Het *profilePath* attribuut van het

gebruikersobject geeft aan of de gebruiker een zwervend profiel heeft. Als dit leeg is, dan is het profiel van de gebruiker lokaal. Het *profilePath* attribuut van de gebruiker kan worden gewijzigd met *dsc.msc*. In de *Profile* tabpagina van de *Properties* van de gebruiker wordt het tekstvak *Profile Path* ingevuld met het UNC pad van de share waar de profielen worden opgeslaan. De fileserver die de share bevat moet geen domeincontroller zijn. De profielen in een submap van de *SYSVOL* share plaatsen, is af te raden.

Roaming profiles worden bij elke in- en uitlogprocedure resp. van en naar de server gekopieerd. Het gebruikers profiel wordt lokaal gecachet op de harde schrijf van de werkpost.

Verder is er ook nog een onderscheid tussen *V1* en *V2* profielen. Ze zijn afkomstig van resp. NT5 en NT6 werkposten, en zijn incompatibel.

6.c Geef de verschillende componenten van gebruikersprofielen.

Het gebruikersprofiel is in te delen in twee belangrijke componenten:

- Het *hivebestand NTUser.dat* bevat de registergegevens van de gebruiker voor diverse windowsinstellingen en toepassingsinstellingen. Het bijbehorende logbestand *NTUser.dat.log* beschermt het hive bestand als er wijzigingen worden weggeschreven naar schijf.

- Het bestandsgedeelte van het profiel bestaat uit diverse submappen en snelkoppelingen. De structuur van V1 en V2 profielen is hier verschillend. Om compatibiliteit met V1 toepassingen te bieden, is de structuur van een V1 profiel afgebeeld op die van een V2 profiel met *junction points*. Om de grootte te beperken, wordt bij voorkeur snelkoppelingen gebruikt, en worden de *Local Settings* (V1) of *AppData\Local* (V2) enkel lokaal gecachet. De cache van internet explorer behoort hier bijvoorbeeld toe.

6.d Over welke alternatieve hulpmiddelen beschikt een beheerder om gebruikersprofielen te configureren?

Om de schijfruimte bezet door de gebruikers profielen vrij te maken, kan worden gebruik gemaakt van de opdracht *delprof.exe*. De optie */d:x* verwijdert de profielen die x dagen inactief zijn geweest. Lokaal opgeslagen en gecachete gebruikersprofielen bekijken kan ook met *sysdm.cpl*. In het tabblad *Advanced* klik je op de knop *Settings* in het *User Profiles* paneel. Hier wordt de grootte en de soort van het profiel aangegeven.

Aangepaste profielen aanmaken, kan door in te loggen als een willekeurige gebruiker, en de instellingen in de desktop omgeving in te stellen zoals gewenst. Een sjabloonaccount kan hierbij nuttig zijn. Om het gebruikersprofiel naar de server te kopiëren kan *robocopy* worden gebruikt. In dit geval moeten de ACL’s op de registersleutels in NTUser.dat manueel worden aangepast met de registry editor. Alternatief kunnen de profielen worden gekopiëerd met het *System Applet* (*sysdm.cpl*, zie hierboven.) door de knop *Copy To* te gebruiken. In het dialoogvenster wordt de nieuwe locatie ingesteld, en de gebruikers die het profiel mogen gebruiken.

De standaardinstellingen van een gebruikersprofiel kunnen worden gemanipuleerd door de *Default User, Default User.V2, Default, All users* en *Public* profielen te manipuleren. De *Default User* en *Default User.V2* profielen op de *NETLOGON* share vormen een uitstekende combinatie met roaming profiles.

Het gebruik van *mandatory roaming profiles* zorgt ervoor dat een gebruiker geen wijzigingen kan opslaan in zijn profiel. Deze profielen kunnen door meerdere gebruikers worden gedeeld. Ze worden bij voorkeur wél opgeslagen in de *SYSVOL* share van een DC. Het *profilePath* van de gebruikers bevat dan de variabele *%logonserver%* die wordt geëxpandeerd tot naam van de domeincontroller die de aanvraag verwerkt. Om een profiel *mandatory* te maken, volstaat het om *NTUser.dat* te hernoemen naar *NTUser.man* en het bestand voor schrijftoegang te beschermen.

Beheerders kunnen afdwingen dat een *mandatory* profile beschikbaar is via netwerk, door de profielmappen de extensie *.man* (NT5) of *.man.V2* (NT6) te geven. In dit geval spreekt men van een *super mandatory profile*.