Robert Fransen is freelance consultant

Windows Vista en applicatiecompatibiliteit

WERKEN ALS STANDARD USER

Windows Vista is in aantocht en brengt veel nieuwe mogelijkheden en verbeteringen. Er is ook een aantal veranderingen doorgevoerd ten behoeve van een hogere veiligheid van de gebruiker en diens computer. Die veranderingen kunnen gevolgen hebben voor de compatibiliteit van applicaties. In dit artikel kijken we naar applicatiecompatibiliteit van Windows Vista.

Bij elke nieuwe versie van een besturingssysteem komt de vraag over applicatiecompatibiliteit naar boven. Het goede nieuws is dat applicatiecompatibiliteit bij Windows Vista hoger is dan bij alle voorgaande Windows-versies. Toch zijn er zaken waar je rekening mee zult moeten houden. Als eerste het bekende probleem rond versienummers. Windows Vista zal versienummer 6.0 teruggeven. Hoewel het probleem rond versienummers een vrij bekende is, blijken er altijd toch nog applicaties de fout in te gaan. Ze doen netjes een versiecontrole bij aanvang, maar verlangen een expliciete Windows-versie. Soms is dit terecht, maar de backward compatibility van Windows Vista is zo goed dat Windows XP-applicaties gewoon zullen draaien. Helaas is de versiecontrole bij sommige applicaties niet goed geïmplementeerd waardoor de applicatie niet start of een error-melding oplevert. Het versieprobleem kan eenvoudig verholpen worden door middel van een versieleugen met behulp van de Program Compatibility Wizard van Windows Vista of met de Compatibility Administrator uit de Application Compatibility Tool Kit.
Met welke echte veranderingen moet je rekening houden? Dagelijks hebben gebruikers last van spyware, malware, virussen en andere opdringerige programma's. Het kan vervelende gevolgen hebben wat betreft de veiligheid, stabiliteit en integriteit van een systeem als gebruikers zelf bewust of onbewust applicaties installeren. Voor bedrijven is dit een grote kostenpost, maar het levert eveneens een groot probleem voor thuisgebruikers. De belangrijkste oorzaak is dat gebruikers met te veel rechten de computer dagelijks gebruiken. Om de Windows desktop beter te beveiligen introduceert Microsoft de volgende zaken in Windows Vista: Windows Resource Protection, User Account Control (UAC) ook wel LUA genoemd en Protected Mode voor Internet Explorer 7.0.

User Access Control

De naam is al een aantal keer veranderd van LUA naar UAP en nu User Account Control, maar het doel is gelijk: elke gebruiker van Windows Vista laten werken met de rechten van 'Standard Users'. Zelfs administrators gaan werken als standard user, pas als het nodig is om een beheertaak te doen worden de admin-rechten gevraagd. Die standard user is eigenlijk niet nieuw en bestaat al vanaf Windows NT 3.1. Er is geen enkele reden om bijvoorbeeld een game- of instant messenger-client te starten met admin-rechten. Toch gebeurt dat heel veel. Waarschijnlijk omdat vroeger in Windows 9x iedereen een beheerder was. Die game of instant messenger-client kan dan systeembronnen aanpassen, heeft volledige rechten in de Windows-directory en kan aanpassingen maken in registry bij HKEY_LOCAL_MACHINE (HKLM). Allemaal ongewenst voor gewone applicaties en een zeer zwak punt wat betreft beveiliging. Helaas doen we dit al jaren zo, omdat het werken met twee accounts omslachtig is; bovendien werken sommige applicaties gewoonweg niet onder standard user. Nieuw in Windows Vista is de Protected Administrator. Ook als je als admin aangemeld bent, wordt de Explorer shell en elk proces gestart met minimale privileges, namelijk die van standard user. Bij het aanloggen wordt het admin-token gefilterd, de privileged SID's worden gedisabled en verhoogde rechten worden ontnomen. Er wordt vanaf dat moment gewerkt met het 'filtered token'. Wanneer de administrator bijvoorbeeld een MMC snap-in start, wordt om toestemming gevraagd, na 'ok' van de admin wordt het volledige token (full token) doorgegeven. Het full token wordt ook wel 'linked token' genoemd. Een grappig voorbeeld van hoe snel iets voor een standard user mis kan gaan, is calc.exe in Windows XP. De instelling van de calculator standard of scientific wordt opgeslagen in HKLM (en wordt dus per computer bepaald en niet per gebruiker). Helaas heeft de standard user in Windows XP daar geen rechten. Het aanpassen van deze functionaliteit kun je gerust aan een gebruiker overlaten, daar heb je echt geen administrator voor nodig. Het is in ieder geval niet handig dat die instelling per machine is en niet per user. In Windows Vista zijn scenario's zoals calc.exe grondig onder de loep genomen en allemaal aangepast zodat werken als standard user ook echt werkt.

User Interface Privilege Isolation

Omdat er nu processen met standard user en met admin-rechten draaien onder een gebruikersaccount is er User Interface Privilege Isolation (UIPI). Dit zorgt dat processen die draaien met admin-rechten (volledig token) worden beschermd om niet te worden misbruikt door de processen die draaien onder standard user (filtered token). UIPI heeft betrekking op het grafische subsysteem bekend als USER. UIPI neemt geen actie of verandert Windows Messaging bij applicaties van hetzelfde niveau. Applicaties met een laag niveau kunnen geen Windows Messages sturen naar applicaties met een hoger niveau, andersom kan dat wel. Een SendMessage of PostMessage lijken succesvol, maar komen nooit aan bij het proces met het hogere niveau. Ook Dynamic Link Library (DLL)–injection van een proces op een lager niveau naar een hoger wordt uitgesloten. Het desktop-window is wel toegankelijk vanuit een lager proces. Het USER/graphics device interface (GDI) model valt niet onder de controle van UIPI. Daarom is het wel mogelijk om vanaf een lager proces schermelementen van een proces met hoger niveau te bewerken.

Windows Resource Protection

Windows Resource Protection (WRP) is het vervolg op Windows File Protection (WFP) van Windows XP. In Windows XP wor-

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<assembly xmlns="urn:schemas-microsoft-com:asm.v1" manifestVersion="1.0">
  <assemblyIdentity version="1.0.0.0"
    processorArchitecture="X86"
    name="AdminApp"
    type="win32"/>
  <description>Beschrijving van de AdminApp</description>
  <!-- Identify the application security requirements. -->
  <trustInfo xmlns="urn:schemas-microsoft-com:asm.v3">
    <security>
      <requestedPrivileges>
        <requestedExecutionLevel
          level="requireAdministrator"
          uiAccess="false"/>
      </requestedPrivileges>
    </security>
  </trustInfo>
</assembly>
```

Codevoorbeeld 1.
Opstartrechten bepalen via AdminApp.exe.manifest

```
#define MANIFEST_RESOURCE_ID 1
MANIFEST_RESOURCE_ID RT_MANIFEST "AdminApp.exe.manifest"
```

Codevoorbeeld 2.
Verwijzing in AdminApp.rc naar manifest

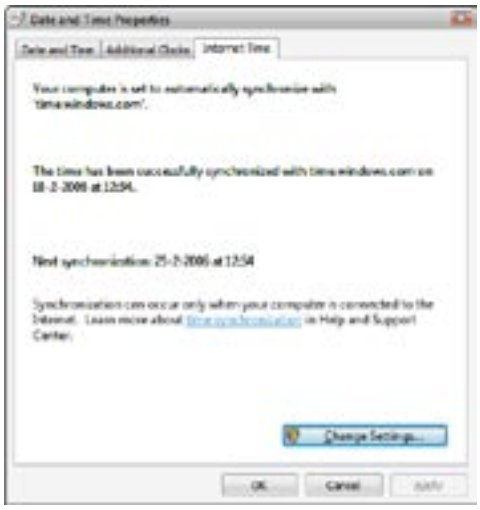
den besturingssysteembestanden beschermd tegen ongeoorloofd overschrijven en verwijderen. Nadat een installatieprogramma (of de gebruiker) besturingssysteembestanden heeft overschreven zet Windows XP die stilletjes weer terug. WRP in Windows Vista daarentegen geeft een 'access denied' terug aan het installatieprogramma als deze een door WRP beschermd bestand wil overschrijven. Dit geldt ook voor de account administrator of system. WRP beschermt niet alleen meer besturingssysteembestanden dan Windows File Protection in Windows XP, maar beschermt nu ook registry-keys. Het distribueren van Windows Vista-binaries met eigen applicaties behoort dus tot het verleden. Alleen de 'OS Trusted Installer Services' kan WRP-resouces updaten.

Virtualization

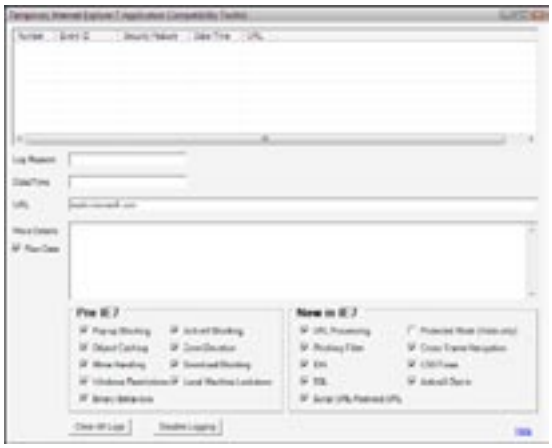
Virtualization in Windows Vista biedt een oplossing voor applicaties die bestanden wegschrijven op plaatsen waar deze eigenlijk niet horen. Omdat in Windows Vista alle applicaties in principe draaien met de rechten van de standard user zullen de volgende bestandslocaties read-only zijn: %SystemRoot% en %ProgramFiles% die meestal verwijzen naar respectievelijk C:\windows\ en C:\Program Files\. Wanneer de applicatie oldapp.exe (met standard user-rechten) het volgende bestand C:\Program Files\oldapp.xyz probeert weg te schrijven, zal dit zonder probleem lukken. Het bestand wordt echter weggeschreven in %LocalAppData%\VirtualStore\Program Files\. Wanneer de gebruiker of de applicatie kijkt in C:\Program Files\ is het bestand oldapp.xyz virtueel te beschikbaar. Wanneer een andere gebruiker op dezelfde machine in C:\Program Files\ kijkt is het bestand niet te vinden. Opgemerkt moet worden dat gebruikersgegevens in Windows Vista in '\Users' geplaatst worden en niet meer in '\Documents and Settings'. Het volledige pad wordt dan:
C:\Users\{*gebruikersnaam*}\AppData\Local\VirtualStore\Program Files\oldapp.xyz
De directory \Program Files is nooit bedoelt als gezamenlijke opslagplaats. Dit heeft wel tot gevolg dat iedere gebruiker dus de hoogste score heeft in een spel, omdat dit niet langer gedeeld wordt onder de gebruikers.
Virtualization is ook actief in de registry voor alle keys onder HKEY_LOCAL_MACHINE>Software. Als een applicatie een key probeert aan te maken onder HKEY_LOCAL_MACHINE>Software wordt deze op de volgende locatie gemaakt: HKEY_CLASSES_ROOT>VirtualStore>Software. Vervolgens is de key virtueel beschikbaar op HKEY_LOCAL_MACHINE>Software voor de applicatie. Virtualization is puur bedoeld voor compatibiliteitsredenen. Nieuwe Windows Vista-applicaties dienen hier geen gebruik van te maken. Tijdens run-time dient een applicatie alleen data te op te slaan in per-user locaties of in de shared documents-directory.

Manifest

Er zijn genoeg applicaties die een legitieme reden hebben om met admin-rechten te draaien. Wanneer je wilt dat een applicatie direct om de admin-rechten vraagt, kun je dit aangeven doormiddel van een manifest. Om bijvoorbeeld een Win32-applicatie genaamd 'AdminApp.exe' te voorzien van een manifest, creëer je een xml-file als in codevoorbeeld 1 en geeft het de naam AdminApp.exe.manifest.
Met *requestedExecutionLevel* kun je het gewenste niveau opgeven. De volgende niveaus zijn mogelijk: *asInvoker*, *highestAvailable*, *requireAdministrator*. Vervolgens maak je een verwijzing in de resource-file naar het manifest, zodat het uiteindelijk in het resource-gedeelte van de PE-file (portable Executable) terechtkomt; zie codevoorbeeld 2. Voor .NET-applicaties moet je het manifest toevoegen in een postbuild-fase aan het resource-gedeelte van de PE-file. Het is wel aan te bevelen om na het toevoegen van een manifest de applicatie te voorzien van een Authenticode-signature.
Veel lastiger wordt het voor control panel-applets. Als een standaardgebruiker onder Windows XP op de klok-applet in het control panel klikt of dubbelklikt op de klok rechtsonder, dan komt de volgende melding: '*You do not have the proper privilege level to change the system time*'. Terwijl veel gebruikers juist even dubbelklikken op de klok om de kalender zichtbaar te maken. Het weergeven van informatie in een control panel-applet is één ding, het aanpassen is iets anders. Er is een goede manier om een control panel-applet beschikbaar te maken voor standaardgebruikers en wanneer een aanpassing gedaan moet worden om admin-rechten te vragen. Een klein schild geeft aan dat admin-rechten nodig zijn voor een bepaalde handeling; zie afbeelding 1. De manier om voor dit soort scenario's een oplossing te bouwen is als volgt. Een control panel-applicatie heeft *asInvoker* in het manifest staan en start als standard user. Deze applicatie start een tweede programma via ShellExecute(). Het tweede programma heeft *requireAdministrator* in zijn manifest staan, dit zorgt er voor dat om een adminstrator-account en password wordt gevraagd. Het tweede programma kan nu de beheertaak uitvoeren. Op deze manier kun je dus splitfunctionaliteit aanbieden; de standard user kan de tijdzone aanpassen, maar niet de tijd of de time server, dat is exclusief voorbestemd voor de beheerder.
Veel programma's doen tegenwoordig automatisch een controle of een nieuwe versie beschikbaar is. Dat is natuurlijk handig, maar je moet er wel voor zorgen dat die update dan ook geïnstalleerd kan worden bij een standard user. Met Microsoft Windows Installer (MSI) 3.1 en 4.0 is het mogelijk om patches te installeren zonder de hele applicatie opnieuw te installeren. Een gebruiker is dan in



Afbeelding 1. De nieuwe datum en tijd control panel-applet.



Afbeelding 2. De Internet Explorer 7 Application Compatibility Toolkit

staat een applicatie die geïnstalleerd is (per-machine) te patchen zonder admin-rechten.

Internet Explorer

Ook op het gebied van Internet Explorer zijn er de nodig veranderingen. Het gaat in dit artikel te ver om daar dieper op in te gaan. Twee opvallende zaken zijn de Protected Mode en dat standaard alle ActiveX-controls op een pagina uitgeschakeld zijn. Op Windows Vista draait Internet Explorer 7 in Protected Mode. Dit is niet beschikbaar voor IE 7 op Windows XP of Windows Server 2003. Protected Mode wil zeggen dat Internet Explorer draait als standard user en User Interface Privilege Isolation dus van toepassing is. Hierdoor is de attack-surface aanzienlijk kleiner. Internet Explorer kan alleen in de tijdelijke internet-bestanden schrijven. Het is mogelijk om logging aan te zetten. In het eventlog worden dan alle sites die een probleem geven gelogd. Ook mogelijke bedreigingen zoals: phishing, spoofing, foute certificaten en dergelijke worden gelogd. Een handig hulpmiddel om compatibiliteitsproblemen op te sporen is de Internet Explorer 7 Application Compatibility Toolkit. Er is nu een 'Temporary'-versie beschikbaar; zie afbeelding 2. Temporary is een ander woord is voor pre-bèta, de tool zal ergens voor de zomer uitkomen. De tool geeft precies aan wat de oorzaak is waarom een bepaalde pagina niet werkt.

Andere zaken

Er zijn nog wat andere zaken waar je misschien rekening mee moet houden. Op het gebied van firewall en antivirus zijn de API's aangepast dan wel nieuwe beschikbaar gekomen. De documentenfolderstructuur is aangepast. De gebruikersdata zitten nu onder c:\users\{*gebruikersnaam*}. De nieuwe gebruikersinterface of beter gezegd de gebruikers-experience codenaam 'AERO' ziet er goed uit, maar kan voor problemen zorgen als jouw applicatie niet goed met themes omgaat. Testen blijft noodzakelijk.

Tools

Er staan heel wat hulpmiddelen tot onze beschikking om compatibiliteitsproblemen op te sporen en te verhelpen.

- Windows Vista Program Compatibility Assistant. Dit is een vast onderdeel van Windows Vista. Hiermee kun je applicaties die ontwikkeld zijn voor een oudere Windows-versie laten werken.
- Windows Application Compatibility Toolkit. Versie 4.1 van deze kit is nu beschikbaar, de opvolger 5.0 is in ontwikkeling en is speciaal voor Windows Vista. Met deze kit kun je een inventaris maken van alle applicaties. Aan de hand van een online database kun je zien welke settings nodig zijn om een bepaalde applicatie te laten draaien. Ook kun je jouw bevindingen op deze manier delen met anderen. Deze tool is zeer uitgebreid en biedt veel meer mogelijkheden dan de Program Compatibility Wizard van Windows Vista zelf.
- AppVerifier 3.0. Deze tool is bedoeld om tijdens de development-fase mogelijke compatibiliteitsproblemen op te sporen. Je kunt met deze tool onder andere foutief gebruik van handles, virtual memory en heap-errors opsporen. Het doel van AppVerifier is ook om applicaties te testen zodat ze voldoen aan het 'Designed for Windows Logo'. Deze tool wordt ook wel LUA Predictor genoemd, omdat je een applicatie kunt testen op mogelijke problemen wanneer de applicatie als standard user wordt gebruikt. Helaas werkt dit onderdeel van AppVerifier niet 100% en rapporteert het veel false positives en false negatives.
- LUA Buglight. En er is een tool in ontwikkeling die binnenkort zal uitkomen onder de naam 'LUA Buglight'. Deze tool is speciaal bedoeld om problemen op te sporen bij gebruik van standard user in Windows Vista. Je start de te testen applicatie vanuit LUA Buglight als standard user. Vervolgens worden alle API'S afgevangen en met het 'filtered token' doorgegeven. Als er een 'access denied'-error komt, probeert LUA Buglight dezelfde call opnieuw, maar dan met admin-rechten (full token). Als dezelfde call dan geen 'access denied'-error geeft, wordt dit als een LUA-bug in het log weggeschreven.
- Upgrade Advisor. Voordat Windows Vista op de markt komt, zal er een Upgrade Advisor beschikbaar komen. Hiermee kunnen thuisgebruikers, maar ook organisaties hun systeem controleren of er nog problemen te verwachten zijn op het gebied van applicatiecompatibiliteit en beschikbaarheid van drivers.
- Regmon en Filemon. Dit zijn twee tools van www.sysinternals.com die zeker aan te bevelen zijn. Beide tools zullen zeker interessante data opleven over wat een bepaalde applicatie allemaal aan het doen is in de registry en het bestandssysteem.
- Internet Explorer Application Compatibility Toolkit. Binnenkort verschijnt deze tool om compatibiliteitsproblemen in webapplicaties met Internet Explorer 7 op te sporen.
- Virtual PC. Een alternatieve manier om een applicatie te draaien die een bepaalde Windows-versie vereist, is om gebruik te maken van Virtual PC. Twee andere oplossingen zijn om de applicatie via Terminal Services of Virtual Server beschikbaar te stellen.

Maak tijd vrij voor testen

Bij elke build van Windows Vista worden meer dan 600 applicaties van 150 verschillende leveranciers getest. Via verschillende early adopter-programma's hoopt Microsoft ook vroegtijdig applicaties op te sporen die problemen geven. Bij het testen van honderden applicaties door Microsoft is gebleken dat onder standard user op Windows XP 56% van de applicaties goed functioneert. Op Windows Vista is dit zelfs 92%. Dit is voornamelijk te danken aan virtualization van de bestanden en de registry, maar je mag er toch niet zondermeer vanuit gaan dat alles probleemloos werkt. Na het rücksichtslos installeren van shareware vanaf een stapel cd-rom's die toevallig op mijn bureau stond, bleek toch dat een grote hoeveelheid van die applicaties problemen gaf. De oproep aan ontwikkelaars is: maak tijd vrij en test je applicaties op Windows Vista.

Robert Fransen is freelance consultant en parttime werkzaam bij Microsoft Nederland in het webteam. Zijn e-mailadres is robert.fransen@rcenter.nl

Referenties

http://blogs.msdn.com/uac/
http://blogs.msdn.com/aaron_margosis/
http://msdn.microsoft.com/library/en-us/dnlong/html/AccProtVista.asp
http://www.microsoft.com/technet/windowsvista/evaluate/feat/uaprot.mspx
http://www.microsoft.com/windows/appcompatibility/
http://www.microsoft.com/technet/windowsvista/deploy/appcompat/acshims.mspx
Windows Vista User Experience Guidelines: http://msdn.microsoft.com/windowsvista/uxguide
Microsoft Application Verifier: http://www.microsoft.com/downloads/details.aspx?FamilyID=bd02c19c-1250-433c-8c1b-2619bd93b3a2
Internet Explorer Application Compatibility Toolkit: http://msdn.microsoft.com/library/en-us/ietechcol/cols/dnexpie/ie7_compat_log.asp