Netz-Weise
Lernen von den Besten.

Workshop Windows Server 2008 R2

Verwaltung und Fehlersuche

Freundallee 13 a
30173 Hannover
www.netz-weise.de

Inhalt

Active Directory Grundlagen

Überblick

Active Directory ist der Name für die Dienste, über den die Benutzerverwaltung in Windows-Netzwerken gesteuert wird. Die Kerntechniken, die dabei eingesetzt werden, sind die Active-Directory Datenbank, in der Informationen wie Benutzer- und Computerdaten, Gruppen und alle möglichen Systemobjekte abgelegt werden, sowie das Kerberos-Schlüsselverteilungscenter, über den die Autorisierung von Benutzern und Computern sichergestellt wird. Außerdem muss in einem Active Directory Umfeld zwingend ein DNS (Domain Name System) implementiert sein, über den ein Computer beim Starten und beim Anmelden von Benutzern die Active-Directory Dienste findet.

Ein Computer, auf dem die Active-Directory Datenbank sowie das Kerberos-Schlüsselverteilungscenter installiert sind und optional auch ein DNS-Server, ist ein sogenannter Domänen-Controller. Alle Daten, die auf diesem Domänen-Controller gespeichert sind, sind Bestandteil der Domäne. Wichtig ist, dass ein Domänen-Controller immer nur genau einer Domäne angehören kann – dies trifft übrigens auch auf Benutzer und Computer zu. Ein Benutzer ist immer Mitglied genau einer Domäne, genau wie ein Computer!

Eine Domäne kann und sollte über mehr als einen Domänen-Controller verfügen, um Ausfallsicherheit zu gewährleisten. Wenn eine Domäne über mehrere Domänencontroller verfügt, replizieren diese den Inhalt Ihrer Datenbank vollständig über alle Domänencontroller hinweg. Es handelt sich hierbei um eine sogenannte Multimaster-Replikation. D.h., dass alle Server schreibend auf die Domänendatenbank zugreifen können. Werden auf mehreren Domänencontrollern gleichzeitig sich widersprechende Änderungen durchgeführt (z.B. das Kennwort eines Benutzers), so sorgen bestimmte Replikationsregeln dafür, dass am Ende immer nur ein Eintrag in der gesamten Domäne gültig ist.

Es können auch mehrere zusammenhängende Domänen eingerichtet werden. Man spricht dann von einem sogenannten Forest, da neue Domänen auf dem Namen der ersten Domäne (der Stammdomäne) aufbauen. Der Name der neuen Domäne beinhaltet den Namen der übergeordneten Domäne plus einen neuen Domänenbestandteil. (s. Abbildung)

Da zwischen Domänen keine Replikation der Benutzer-und Computerkonten stattfindet, gibt es einen Mechanismus namens Vertrauensstellung, der es Benutzern unterschiedlicher Domänen erlaubt, Berechtigungen auch in anderen Domänen als der eigenen erhalten zu können. Eine Vertrauensstellung ist quasi eine Leseberechtigung auf die Active-Directory-Datenbanken aller Domänen im Forest. Diese Leseberechtigungen sind notwendig, damit ein Server in Domäne Deutschland.netz-weise.de, der einem Benutzer aus der Domäne USA.Netz-Weise.de Berechtigungen vergeben soll, auch die Benutzerkonten aus der Domäne USA anzeigen und diesen z.B. auf Freigaben Berechtigungen vergeben kann. Denn einem Benutzer, den ich nicht kenne, kann ich natürlich auch keine Berechtigungen geben. Das Kerberos-System sorgt dann mit sogenannten Tickets für die Authentifizierung zwischen den Domänen.

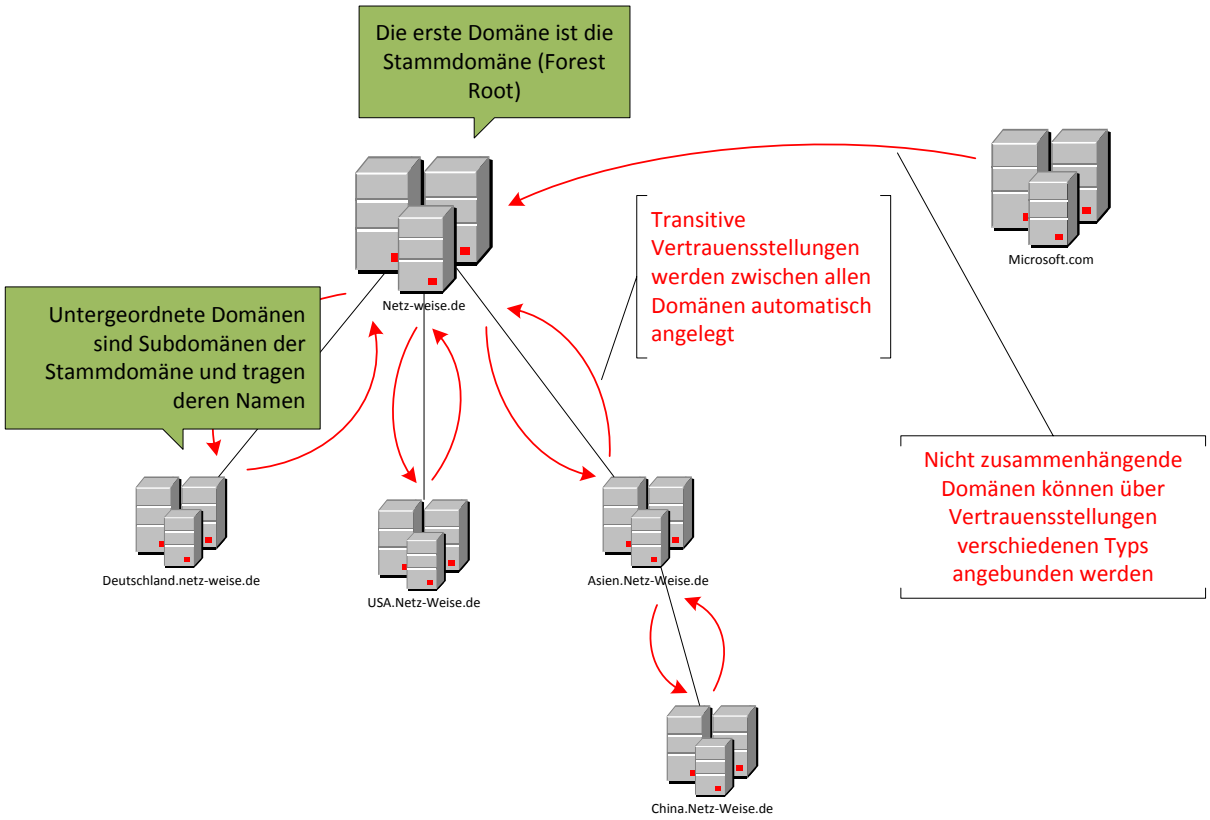


Abbildung 1: Überblick über das Active Directory Domänenmodell

Benutzerkonten und Gruppen

Die Domänencontroller verwalten in Ihren Domänendatenbanken in erster Linie Informationen über Benutzerkonten, Computerkonten und Gruppen. Diese werden zur Anmeldung benötigt. Ein Computer bzw. Benutzer, der kein Konto in der Domäne hat, kann nicht authentifiziert werden.

Beim Anlegen eines Benutzers sind eine Reihe von Informationen wie der NT4-Anmeldename (Domäne\Benutzername), ein Kennwort und eine Reihe von Kennworteigenschaften Pflicht. Tatsächlich hat das Benutzerkonto aber eine ganze Reihe von Informationen, die es verwaltet, die weit über die Anmeldeinformationen hinausgehen. Active Directory stellt nämlich nicht nur Anmeldefunktionalität zur Verfügung, sondern als vollwertiger Verzeichnisdienst können auch eine Reihe Metainformationen verwaltet werden, wie Adressinformationen, Telefonnummern, Abteilungszugehörigkeiten, Hierarchien (Manager) usw. Diese Informationen können mit einer Software abgefragt werden, die die LDAP-Abfragesprache beherrscht (LDAP-Queries). LDAP oder Lightweight Directory Access Protocol ist ein herstellerunabhängiges Protokoll, um Daten in Verzeichnisdiensten wie Active Directory, Open LDAP, Novell NDS usw. zu speichern.

Um Zugriff auf das Netzwerk zu bekommen, muss sich ein Benutzer zuerst an einem Computer anmelden, der selbst Mitglied einer Domäne im Forest ist – was nichts weiter heißt, als dass der Computer ebenfalls ein „Benutzerkonto“ in einer Domäne hat. Nur wenn der PC Mitglied der Domäne des Benutzers oder einer vertrauten Domäne ist, kann der Benutzer diesen PC verwenden, um sich an der Domäne anzumelden.

Der Anmeldevorgang

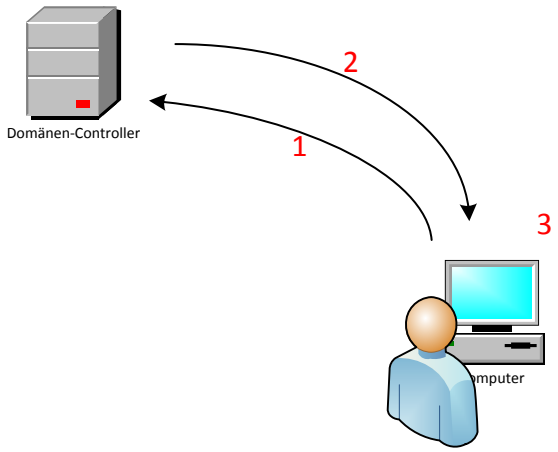
Wenn ein Benutzer eine Anmeldung startet, muss er zuerst seinen Benutzernamen und sein Kennwort eingeben. Ein Benutzername besteht grundsätzlich aus der Domäne des Benutzers, in der

sich der Benutzer befindet, plus seinem Benutzernamen : *deutschand\Holger*. Wir kein Domänenname angegeben, verwendet der Computer einen Standard. Dieser kann unterschiedlich sein, normalerweise verwendet der Computer jedoch seine eigene Domäne. Alternativ kann man einen sogenannten UPN (User Principal Name) verwenden, der Forestweit eindeutig ist und daher keinen zusätzlichen Domänennamen benötigt. Dieser UPN sieht aus wie eine Email-Adresse und kann, je nach Konfiguration, auch dieser entsprechen: `Holger@netzweise.de`.

Wenn der Benutzer seine Anmeldeinformationen eingibt, schickt der Computer diese zur Überprüfung an den Domänencontroller. Hierbei wird jedoch nie das Kennwort selber übertragen. Sind der Benutzername und das Kennwort korrekt, schickt der Domänencontroller die Informationen über sämtliche Gruppenmitgliedschaften an den PC zurück, und dieser erstellt aus diesen Informationen eine Art Personalausweis (Access Token), dass alle wesentlichen Sicherheitsinformationen enthält. Dieses Access-Token kann wird jetzt bei jedem Zugriff auf eine Resource auf dem PC verwendet, um zu prüfen, ob ein Benutzer in den korrekten Gruppen ist, um Zugriff zu erhalten.

Wichtig! Da das Access-Token nur bei der Anmeldung erstellt und danach nicht aktualisiert wird, bis der Benutzer sich neu angemeldet hat, werden Änderungen an Gruppenmitgliedschaften eines Benutzers erst mit einer Neuanmeldung gültig! Alle anderen Änderungen wie NTFS-Berechtigungen sind sofort wirksam, aber das Hinzufügen oder Entfernen von Rechten und Berechtigungen über Gruppenmitgliedschaften erfordert immer eine Neuanmeldung des Benutzers!

Um sich die Gruppenmitgliedschaften des angemeldeten Benutzers anzeigen zu lassen, kann man den Befehl `whomai /groups` verwenden. Er liest das Access-Token aus und gibt eine Auflistung aller Gruppen an.



1. Der PC gibt die Benutzerinformationen an den DC
2. Der DC verifiziert die Benutzerdaten und schickt ein Ticket mit den Benutzergruppen zurück
3. Der erstellt ein lokales (!) Access Token, das für alle lokalen (!) Zugriffe des Benutzers zur Berechtigungsüberprüfung verwendet wird.

Gruppen

Ein wesentliches Konzept bei der Berechtigungsvergabe von Windows sind Gruppen. Gruppen fassen Benutzer zu Einheiten zusammen, und diesen Einheiten kann man dann Berechtigungen vergeben, anstatt jeden Benutzer einzeln zu berechtigen. Die Vorteile liegen auf der Hand:

- Die Dokumentation von Berechtigungen ist deutlich einfacher
- Das Entfernen von Berechtigungen beschränkt sich darauf, den Benutzer aus einer Gruppe zu entfernen

- Das Erweitern von Berechtigungen beschränkt sich auf das Hinzufügen von Benutzern zu Gruppen, anstatt jedes Objekt einzeln anzufassen, auf das Berechtigungen vergeben werden sollen.

Das beste Beispiel für die Nützlichkeit von Gruppen liefert Microsoft selber. In dem Moment, in dem ein Computer in eine Domäne aufgenommen wird, wird nämlich die Gruppe der `Domänen-Benutzer (Domain Users)` Mitglied der Gruppe Benutzer auf dem PC, sowie die Gruppe der `Domänen-Administratoren (Domain Admins)` Mitglied in den lokalen Administratoren. Dadurch wird jeder Benutzer, der in der Domäne angelegt wird, automatisch berechtigt, den PC zu benutzen, da ein neuer Benutzer in der Domäne automatisch Mitglied der `Domain Users` wird. Genauso hat jedes Mitglied der `Domain Admins` automatisch Administrative Rechte auf allen Workstations und Servern der Domäne.

Windows unterscheidet grundsätzlich 4 Typen von Gruppen:

Typ	Beschreibung	Speicherort
Lokale Gruppen	Werden in der lokalen Benutzerdatenbank (SAM = Security Accounts Manager) gespeichert. Eine lokale Gruppe existiert jeweils nur auf dem PC selbst und kann auf anderen PC´s nicht zur Berechtigungsvergabe verwendet werden. Viele lokale Gruppen werden bereits bei der Windows Installation vom System mit Standard-Berechtigungen angelegt wie z.B. Benutzer, Administratoren, Hauptbenutzer, Server-Adnministratoren…	PC
Lokale Domäne	Wie lokale Gruppen, aber nicht nur auf einem PC, sondern innerhalb der gesamten Domäne sichtbar, allerdings nicht in vertrauten Domänen. Den lokalen Gruppen sollen direkt Berechtigungen und Rechte vergeben werden.	AD
Globale	Standardgruppe, um Benutzerkonten hinzuzufügen. Im Unterschied zu lokalen Domänengruppen können Sie nur Benutzer und globale Gruppen aus der eigenen Domäne als Mitglieder haben, sind aber in allen vertrauten Domänen sichtbar und können also auch in Fremddomänen für die Berechtigungsvergabe verwendet werden	AD
Universale	Wie globale Gruppen, können aber Mitglieder aus allen Domänen(!) aufnehmen. Globale Gruppen werden auf einem speziellen Domänencontroller, dem globalen Katalog, gespeichert.	AD

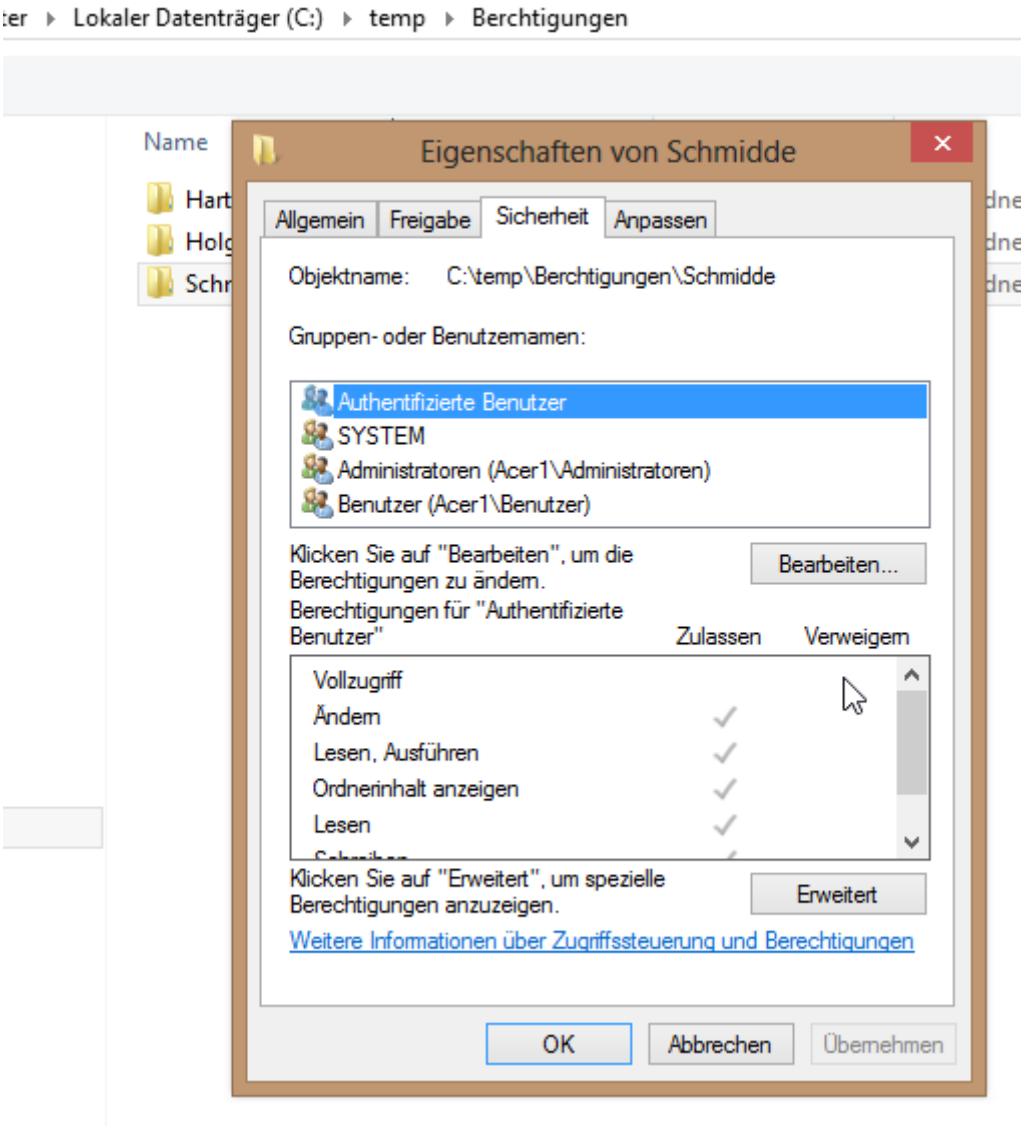
Das von Microsoft empfohlene Konzept zur Berechtigungsvergabe lautet, Benutzer-Konten in globale Gruppen hinzuzufügen, und diese dann lokalen bzw. Domänen-lokalen Gruppen. Berechtigungen werden nur an die beiden Typen von lokalen Gruppen direkt vergeben. Dieses Konzept wird auch mit A-G-P bzw. A-G-DL-P abgekürzt. Hierbei steht A für Account (Benutzer-oder Computerkonto), G für global Group, DL für Domain Local Group, L für Local Group und P für Permission.

Dateisystem-Berechtigungen / Netzwerk-Freigaben

Alle Versionen des Windows-Betriebssystems enthalten eine Sicherheitssystem, das die Möglichkeiten eines Benutzers, sein System zu verändern, über Berechtigungen und Rechte steuert. Windows Rechte sind dabei die globalen Rechte, die von Windows überprüft werden und festlegen, was ein Benutzer mit dem Betriebssystem machen darf, wie z.B. sich an- und ab zu melden, oder sich per Remote Desktop mit dem PC zu verbinden. Berechtigungen steuern, auf welche Dateien ein Benutzer zugreifen darf (und auch auf welche Drucker und welche Active Directory Objekte).

Um den Zugriff auf Dateien zu steuern, ist es zwingend notwendig, dass die Festplattenpartitionen mit NTFS formatiert sind, da nur NTFS in seinen Datei-Metainformationen Benutzerrechte vorgesehen hat.

Berechtigungen im NTFS-Dateisystem können auf Dateien und Ordnern vergeben werden. Dazu wählt man aus dem Kontextmenü den Eintrag Sicherheit (Security) aus:



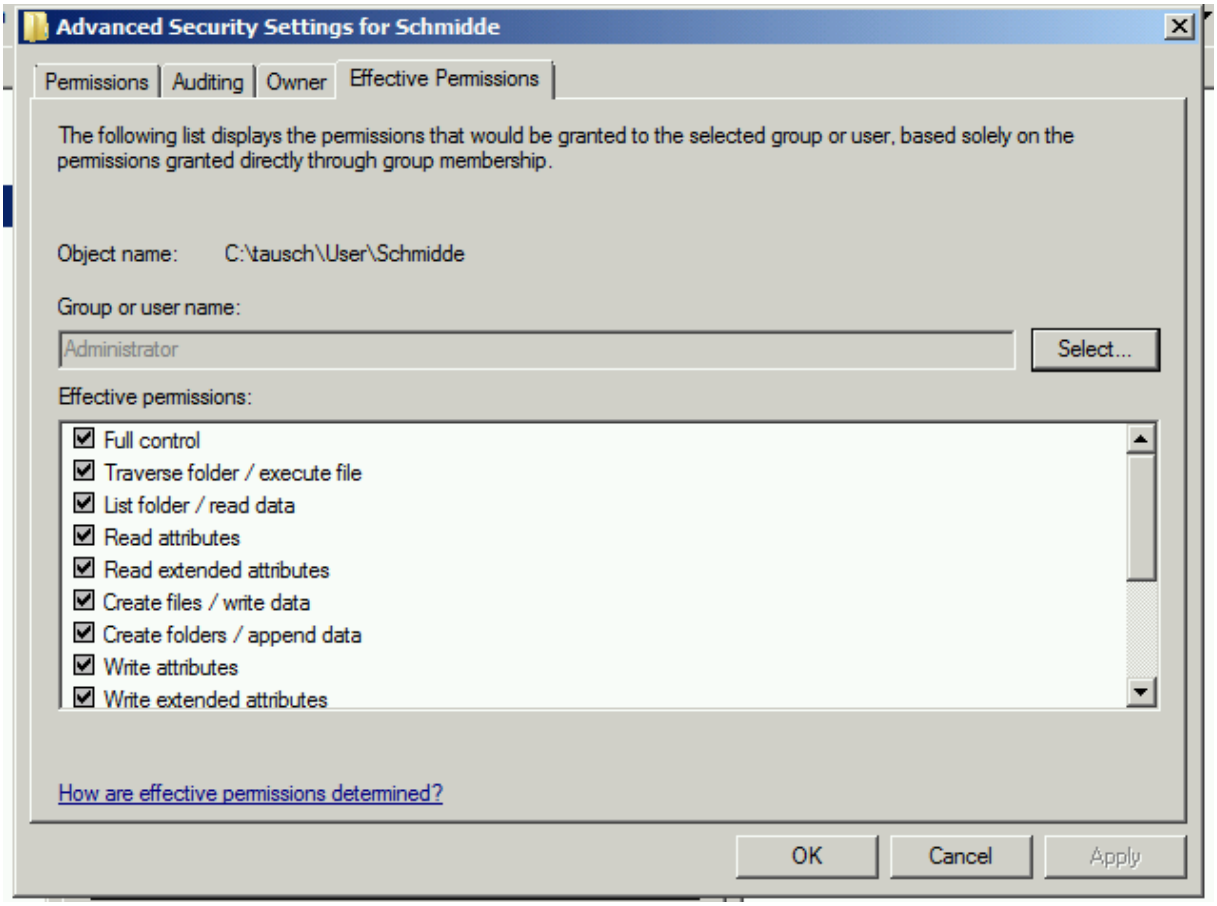
Selbst wenn man eine neue Datei anlegt, sind hier schon eine Reihe von Einträgen gesetzt. Diese Einträge kann man auch standardmäßig nicht entfernen – sie sind vom übergeordneten Objekt ererbt. Quelle aller Berechtigungen ist das Root des Laufwerks, als im Beispiel das Laufwerk C:

Die Standardberechtigungen im NTFS sind:

Lesen	Erlaubt das Lesen von Dateien und Ordnern
Lesen, Ausführen	Beinhaltet Lesen und Ordnerinhalte auflisten und zusätzlich das Starten von ausführbaren Dateien
Schreiben	Erstellen und Beschreiben von Dateien und Ordnern
Ordnerinhalte auflisten	Immer in Kombination mit Lesen, Ausführen erlaubt es das Anzeigen von Ordnerinhalten
Ändern	Lesen, Schreiben, Ausführen und Löschen von Daten
Vollzugriff	Erlauben neben Ändern auch das Vergeben von Rechten

Eine Besonderheit ist das Verweigern von Berechtigungen. Normalerweise sind Rechte additiv – ist ein Benutzer Mitglied mehrerer Gruppen, hat er effektiv die Summe der Rechte aller seiner Gruppen. Die einzige Ausnahme von dieser Regel ist das Verweigern (Deny). Hat ein Benutzer auf einem Recht ein Deny (s. Abb.), so überschreibt dieses Deny alle anderen vergebenen Rechte und der Benutzer oder die Gruppe bekommt diese Berechtigung auf keinen Fall!

Um die effektiven Berechtigungen eines Benutzers zu sehen, kann man im Sicherheits-Menü auf „Advanced“ klicken und dann den Reiter „Effective Permissions“ auswählen. Hier kann man die Rechte sehen, die der Benutzer inklusive aller Gruppen hat, in denen er Mitglied ist.



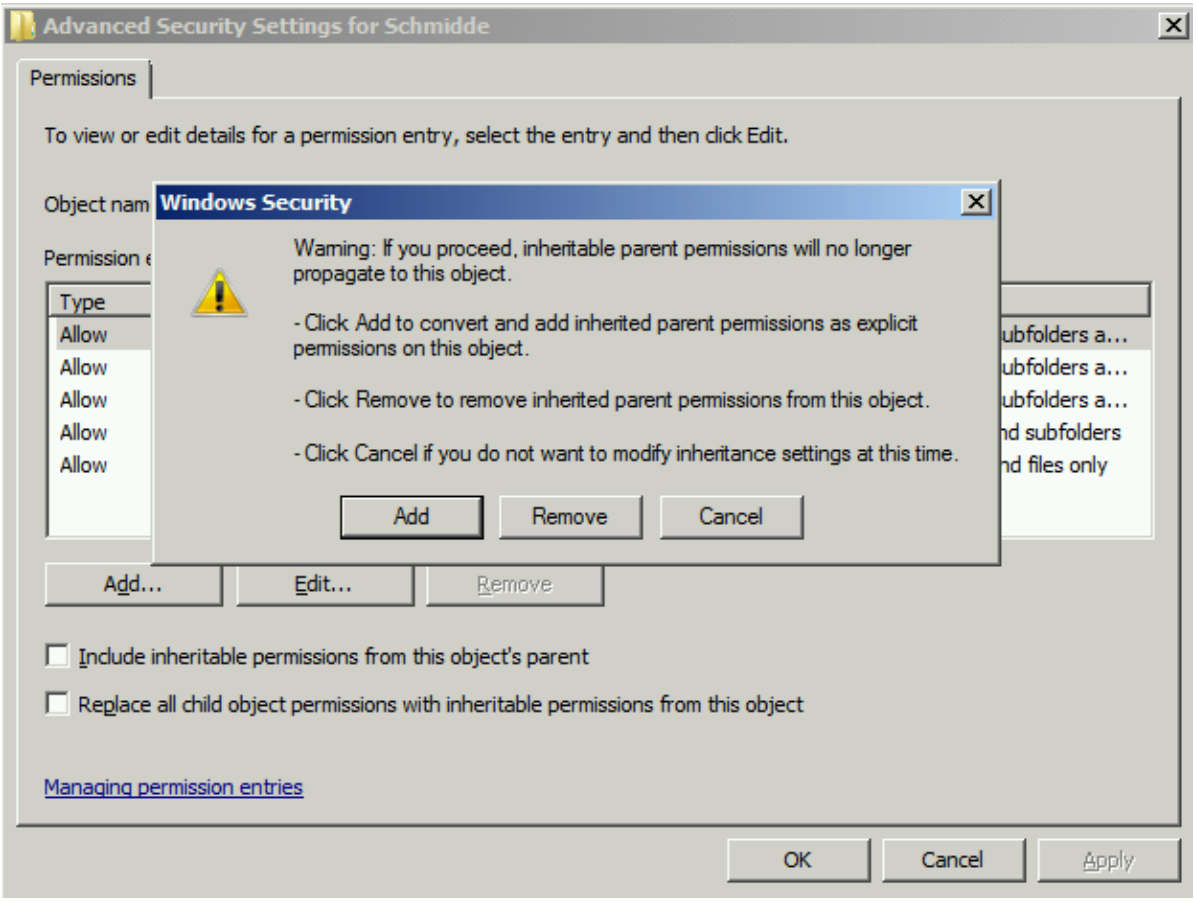
Ein weiteres wichtiges Konzept bei NTFS ist der Besitzer. Der Besitzer (Owner) einer Datei oder eines Ordners kann auch ohne jegliche Berechtigung die Berechtigungen ändern! Um den Besitz an einer Datei zu übernehmen, benötigt man Vollzugriff. Alternativ kann auch ein Administrator den Besitzer einer Datei ändern.

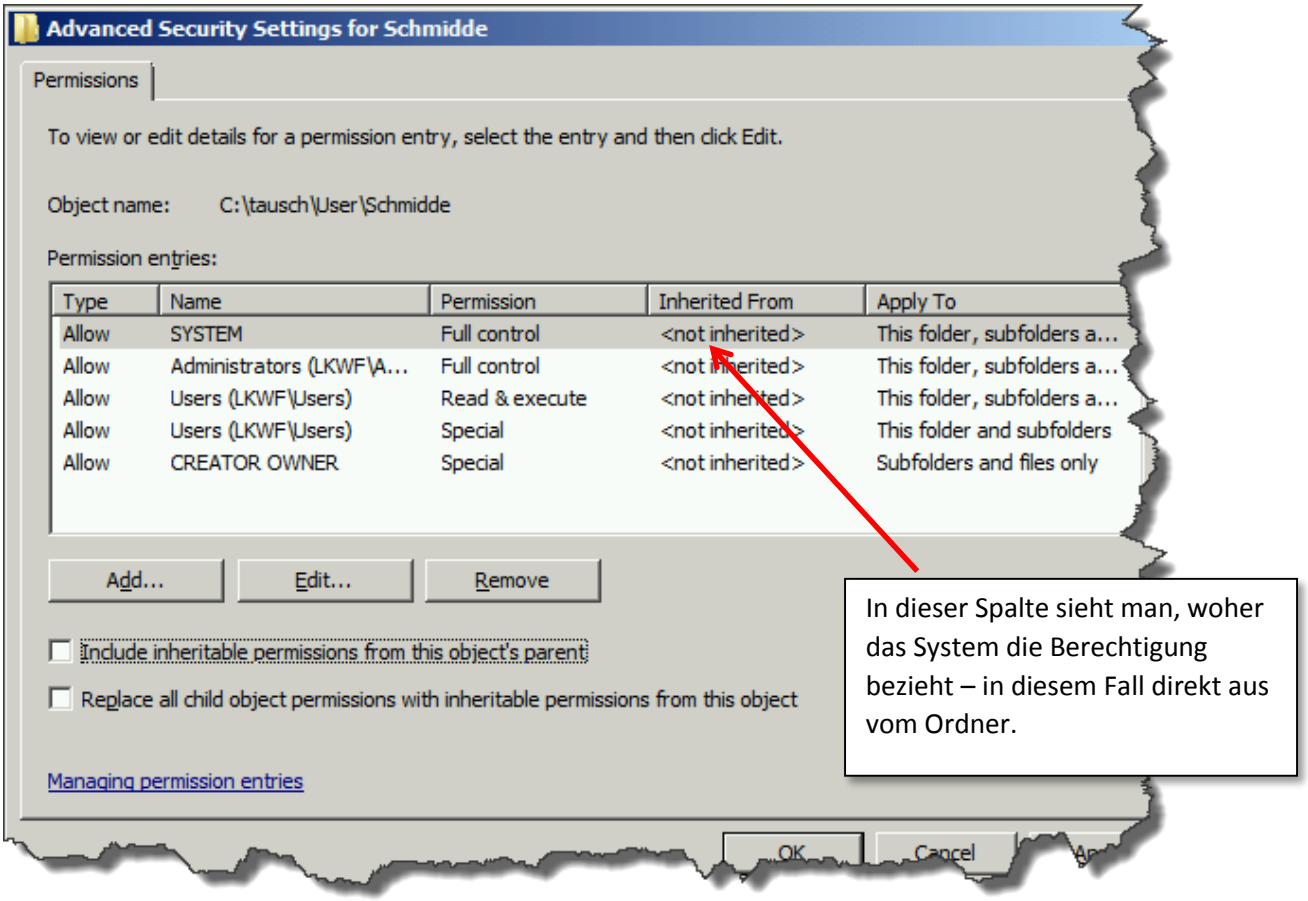
Administratoren haben immer das Recht, den Besitz an einer Datei zu übernehmen, auch wenn sie keinerlei Zugriff auf eine Datei haben!

Vererbung deaktivieren

Wenn man Berechtigungen entfernen möchte, die aufgrund ererbter Berechtigungen nicht änderbar sind, muss man die Vererbung auf dem Ordner oder der Datei abschalten. Dies führt nicht zu einer globalen Deaktivierung der Vererbung, sondern nur zum Unterbrechen der Vererbungskette. Unterhalb des Odners, auf dem man die Vererbung deaktiviert hat, werden die Berechtigungen jetzt wieder ganz normal weitergegeben, nur dass der Ordner, auf dem die Vererbung deaktiviert ist, jetzt als neue Quelle der Berechtigungen gilt.

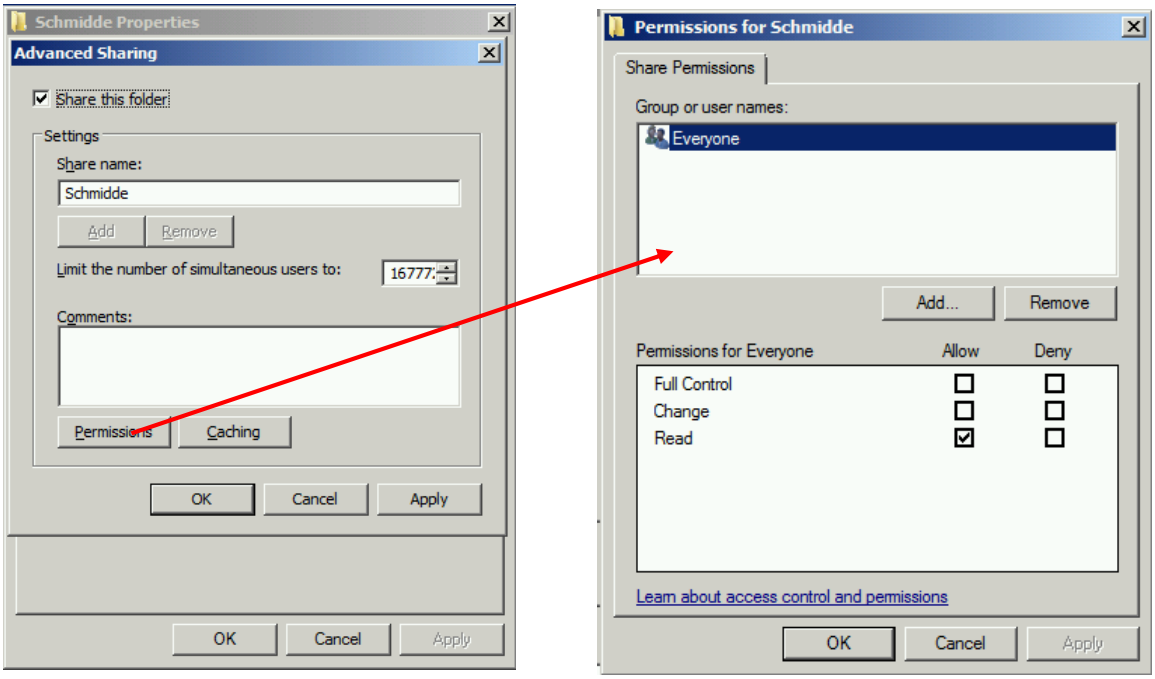
Zum deaktivieren müssen Sie wieder in die erweiterten Berechtigungen gehen (Advanced-Button), und dann auf Change permissions:





NTFS und Freigaben

Wenn man Daten über das Netzwerk verfügbar machen will, muss man diese freigeben. Freigeben bedeutet, dass man dem Windows-Serverdienst, der die Daten über das Netzwerk zur Verfügung stellt, einen Ordner vorgibt, den er über das Netzwerk ausliefern darf. Auch der Windows Serverdienst verwaltet Berechtigungen:



Die Berechtigungen sind hier wesentlich einfacher gehalten als beim NTFS. Es gibt nur Lesen (inkl. Ausführen), Ändern (inkl. Löschen) und Vollzugriff.

Welche Berechtigungen gelten jetzt aber, wenn ein Ordner sowohl NTFS-Berechtigungen als auch Freigabe-Berechtigungen vergeben hat? Die Lösung ist tatsächlich sehr trivial: Die Berechtigungen werden komplett unabhängig voneinander geprüft, einmal vom Server-Dienst (bei der Verbindungserstellung), und einmal vom NTFS (beim Zugriff auf die Daten). Wenn einer der beiden den Zugriff verweigert, kann der Benutzer die gewünschte Aktion nicht durchführen. Hat also der Benutzer z.B. Vollzugriff im Dateisystem, aber nur Lesen-Zugriff auf der Freigabe, kann er über die Freigabe keine Daten ändern oder schreiben, obwohl er laut NTFS alles darf. Wir merken uns also: Effektiv hat der Benutzer über das Netzwerk nur die Rechte, die ihm sowohl per NTFS als auch von der Freigabe gewährt werden. Meldet sich der Benutzer aber lokal am Rechner an, werden die Freigaberecht gar nicht geprüft!

Lokale Sicherheitsrichtlinien

Die lokalen Sicherheitsrichtlinien legen eine Reihe von Sicherheitseinstellungen für Windows fest. Hier finden sich z.B. die Kennwort-Richtlinien, die Systemrechte der Benutzer, eine Reihe von Systemeinstellungen, Firewallkonfiguration und Netzwerk-Zugriffsregeln, IP-Sec-Konfiguration und die Möglichkeit, die Ausführung von Software zu verhindern (Software-Restriction-Policies und App-Locker).

Die Lokalen Sicherheitsrichtlinien können zentral über das Active Directory mit Hilfe der Gruppenrichtlinien überschrieben werden. Alle Einstellungen, die in den Lokalen Sicherheitsrichtlinien festgelegt sind, gibt es in den Gruppenrichtlinien ebenfalls. Wird ein Computer in die Domäne aufgenommen, werden die Kennwortrichtlinien beispielsweise sofort nur noch aus der Domäne bezogen und die lokalen Richtlinien werden überschrieben. Die meisten anderen Richtlinien sind jedoch in der Domäne standardmäßig nicht konfiguriert, so dass die lokalen Richtlinien trotzdem Gültigkeit haben.

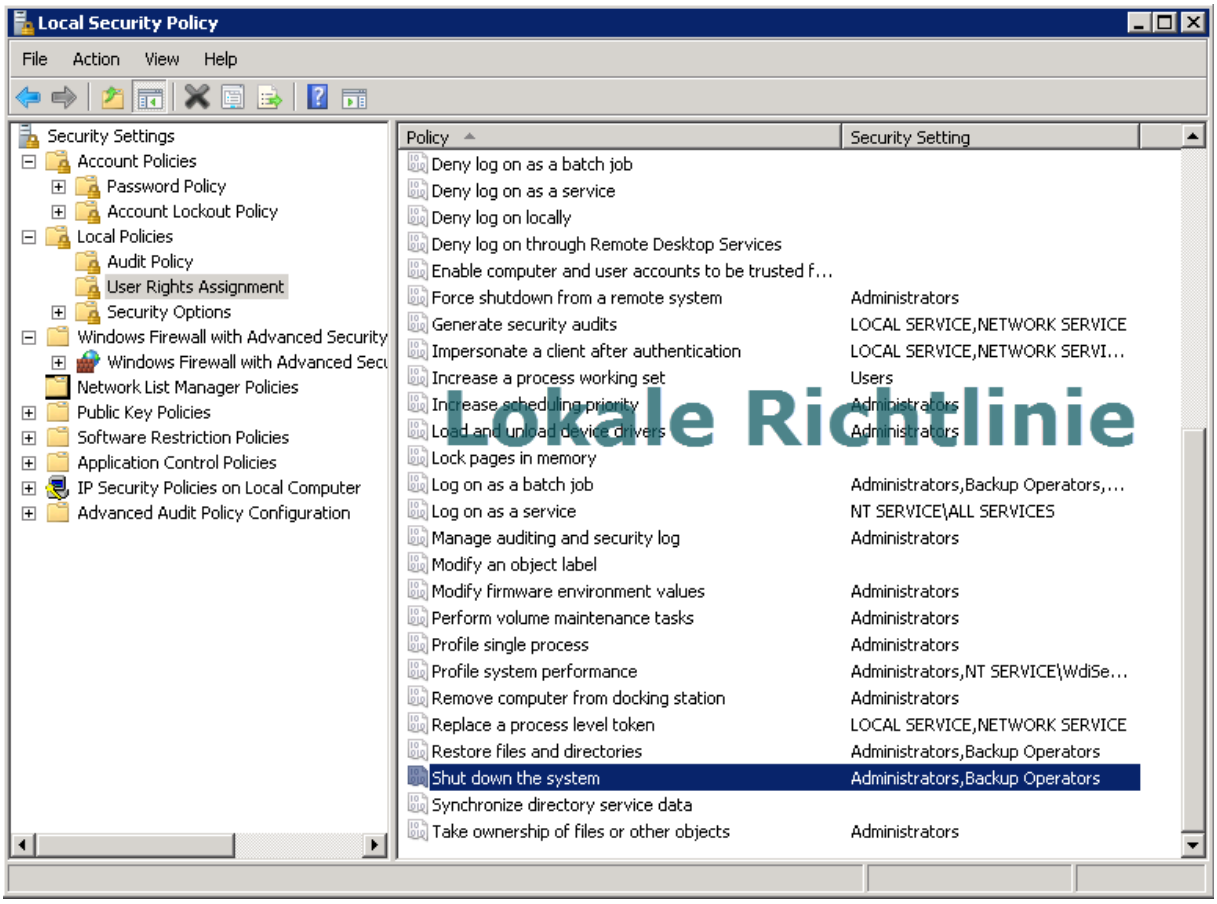


Abbildung 2- Local Security Policy Editor

Für die Konfiguration der lokalen Sicherheitsrichtlinien gibt es eine eigene Konsole „Local Security Policy“ oder „lokale Sicherheitsrichtlinien“ in der deutschen Version. In der Konsole befinden sich die Einstellungen in einzelnen Konfigurationsknoten oder – bildlicher gesprochen - Unterordnern.

Im Ordner *Local Policies -> User Rights Assignment* befinden sich alle Windows Systemrechte aufgelistet. Hier findet man z.B. die Rechte, die einem Benutzer das Anmelden per Remote Desktop (RDP) erlauben – *allow log on through Remote Desktop Services* oder die Rechte, den Computer herunter zu fahren – *Shut down the system*.

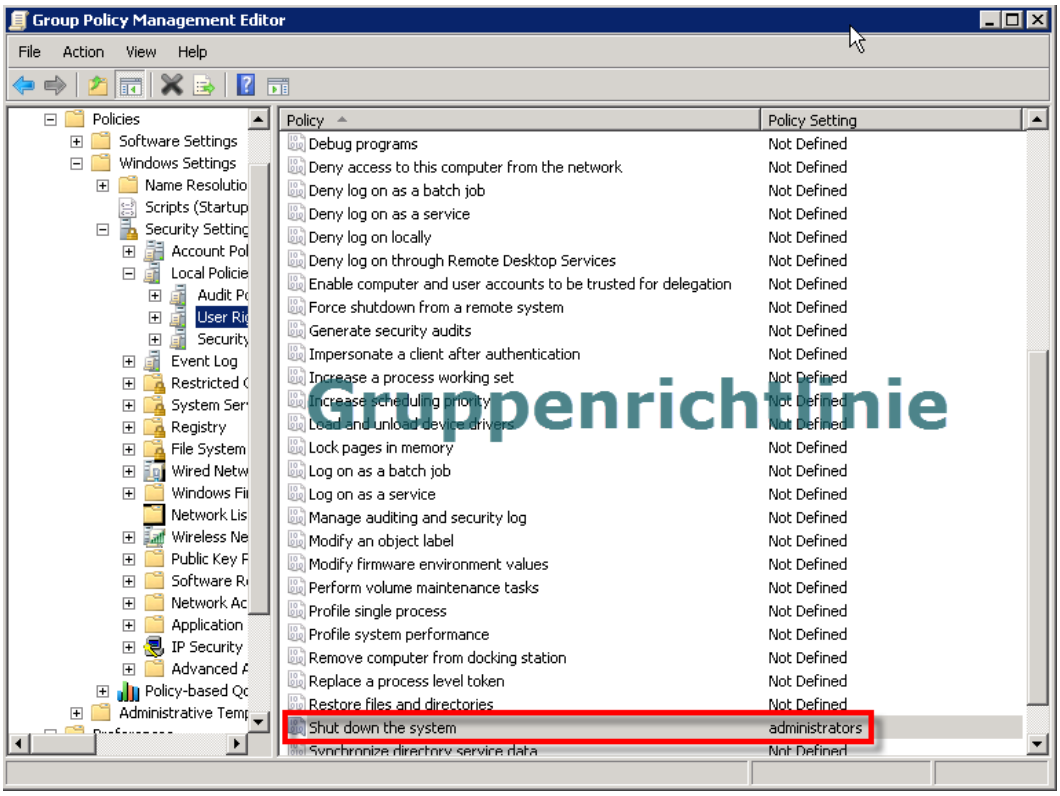


Abbildung 3- es wird eine Gruppenrichtline konfiguriert, die das Herunterfahren nur Administratoren erlaubt

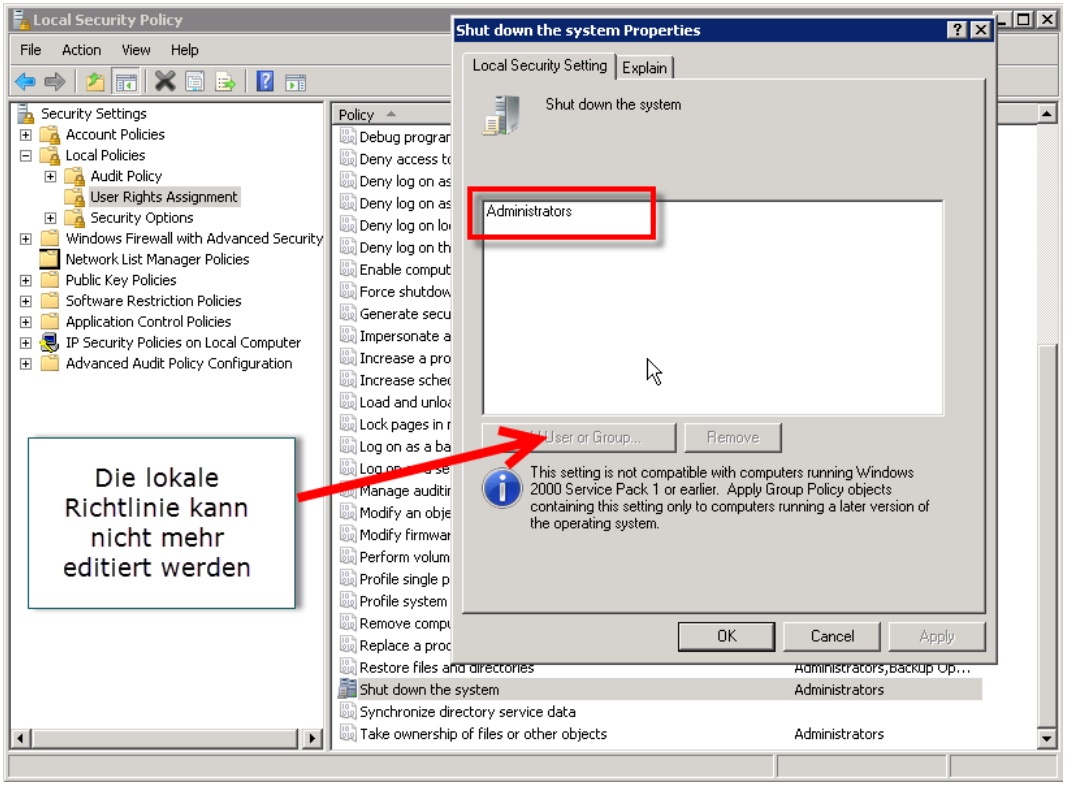


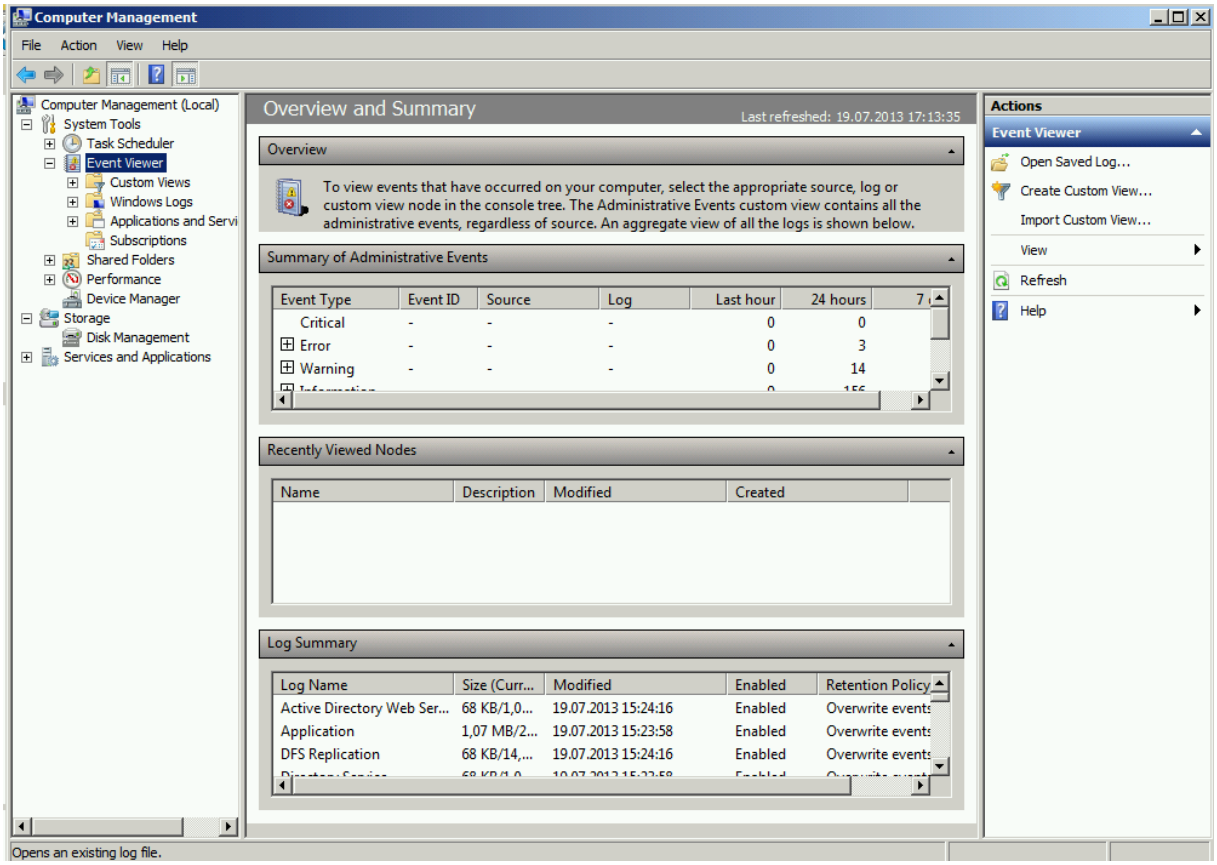
Abbildung 4 - Die Gruppenrichtline hat die lokalen Einstellungen (s. Abb. 2) überschrieben. Eine Änderung ist lokal nicht möglich

Das Windows Eventlogging

Die zentrale Stelle für Verwalten von Systemmeldungen ist die Windows Ereignisanzeige. Dies ist die Stelle, an der alle Microsoft-eigenen Programme ihren Ablauf protokollieren können.

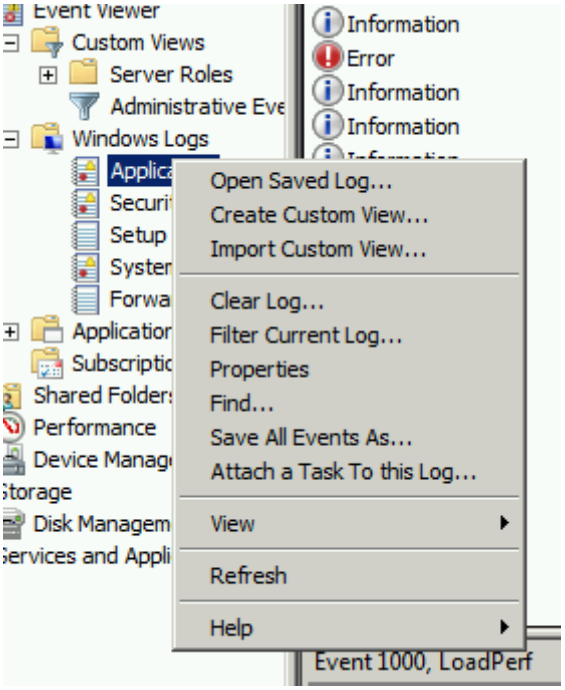
Mit Windows Vista / Windows Server 2008 hat Microsoft die Ereignisanzeige deutlich erweitert. Dies soll den Wildwuchs von Logs an allen möglichen Stellen des Systems entgegen wirken. Grundsätzlich kann mit dem erweiterten Event-Logging jede Software jetzt das Windows Ereignisprotokoll verwenden, um Ablaufdaten zu speichern. Dafür werden die Protokoll in Binärdateien mit der Endung evtx im Ordner `%systemroot%\system32\Winevt\Logs\` gespeichert, wobei %systemroot% für den Windows-Stammordner steht.

Um die Ereignisprotokolle zu lesen und zu verwalten, stellt Windows den Eventlog-Viewer zur Verfügung, der am einfachsten über die Computerverwaltungs-Konsole (compmgmt.msc) zu erreichen ist.

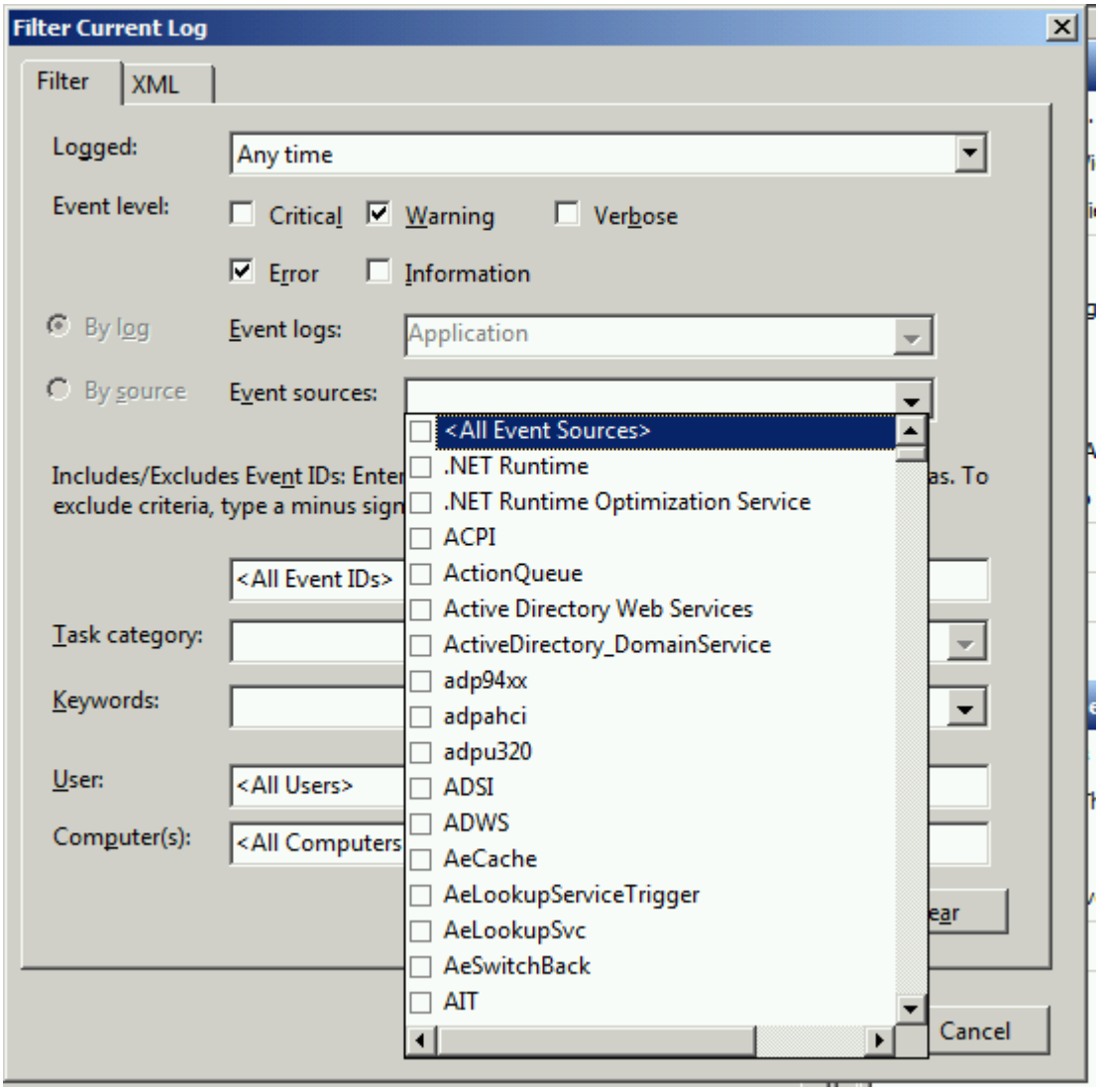


Hier fällt als erstes auf, dass man in der Hauptansicht nicht mehr die 3 Standard-Windows-Event-Logs sieht (Application, Security, System), sondern eine Auflistung der wichtigsten Meldungen aus allen Protokollen und ein Log Summary. An oberster Stelle im Viewer stehen außerdem die „Custom Views“, nicht mehr die Windows Logs. Hintergrund hierzu ist, dass die Anzahl der Logs seit Wndows 2008 durch das neue Logging-System enorm gestiegen ist (schauen sie mal unter „Applications and Services“ nach!). Mit den Custom Views kann man hier Filter definieren, mit denen man sich die wichtigsten Ereignisse wie z.B. alle Error direkt anzeigen lassen kann, ohne alle Logs einzeln durchsuchen zu müssen.

Um Logs nach Fehlern zu durchsuchen und sie zu speichern, gibt es einige Möglichkeiten:

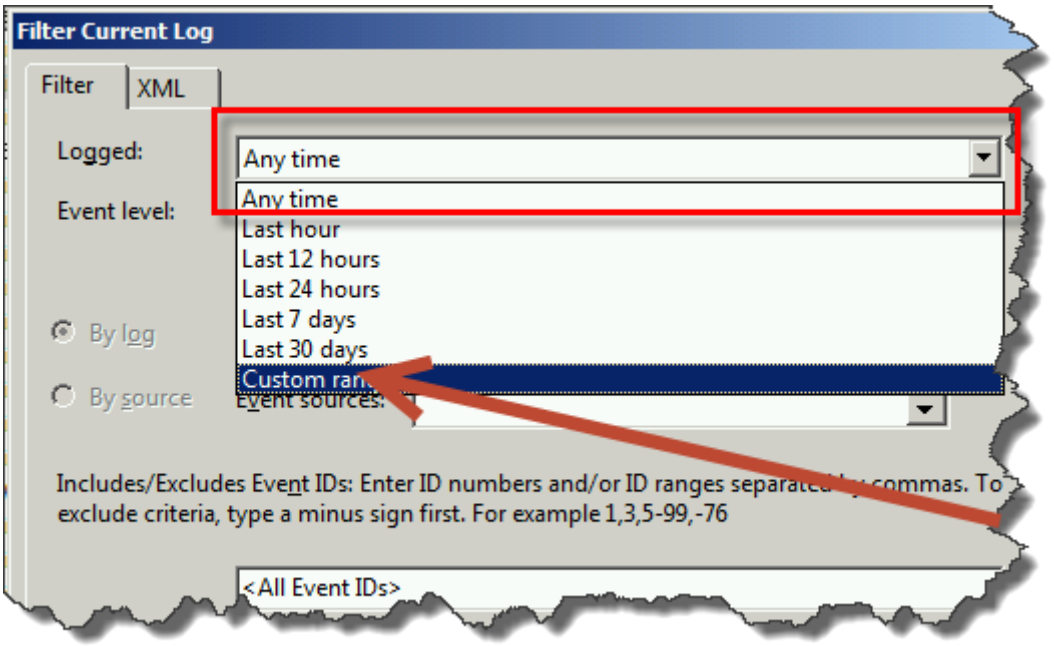


Sie können die Logs Filtern:

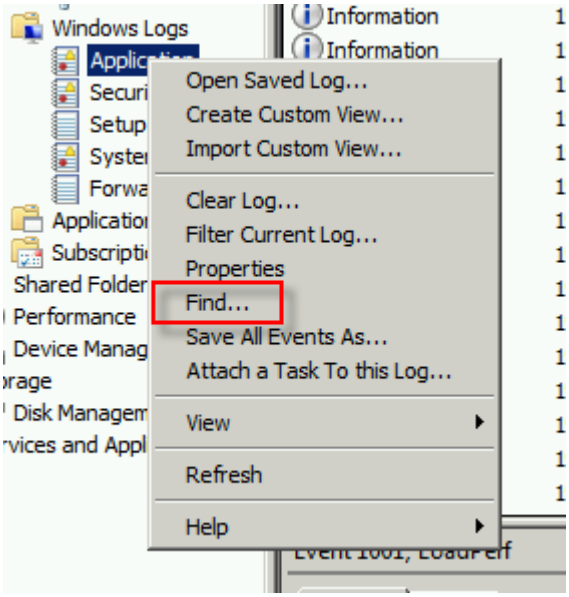


Event-Level gibt an, ob sie Fehler, Warnungen oder Informationen sehen wollen. Informationen sind für die erste Suche oftmals nicht so wichtig. Außerdem können Sie die Quelle angeben (den Dienst, der den Fehler geschrieben hat), oder nach bestimmten Event-IDs. Fast jedes Event ist im System unter eine bestimmten eindeutigen ID registriert. Durch die recht eindeutige Kennzeichnung Quelle + ID ist es oftmals im Internet sehr einfach, die Ursache für den Meldung zu finden. Ein guter Startpunkt ist z.B. EventID.net.

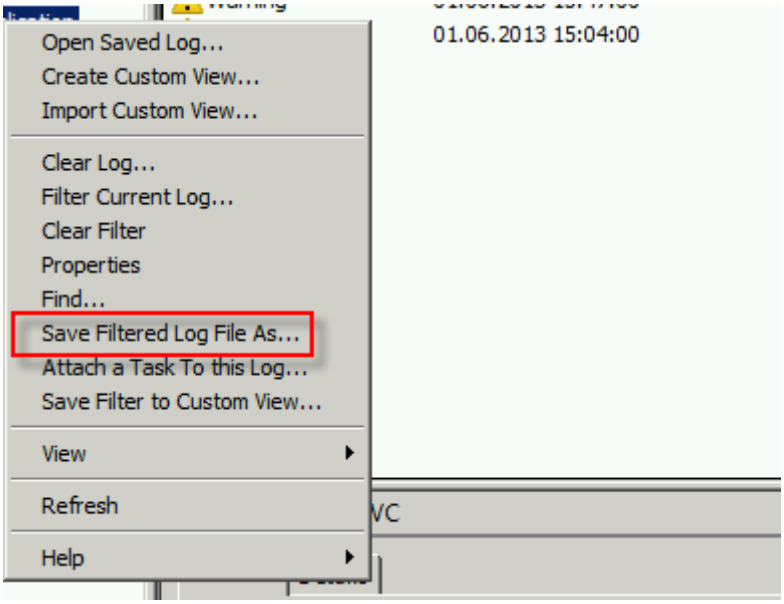
Wenn Sie wissen, zu welchem Zeitpunkt ein Fehler ungefähr aufgetreten ist, können Sie auch einen Zeitrahmen filtern:



Und mit dem Eintrag Find im Kontextmenü eines Protokolls können Sie auch direkt nach Freitext suchen:

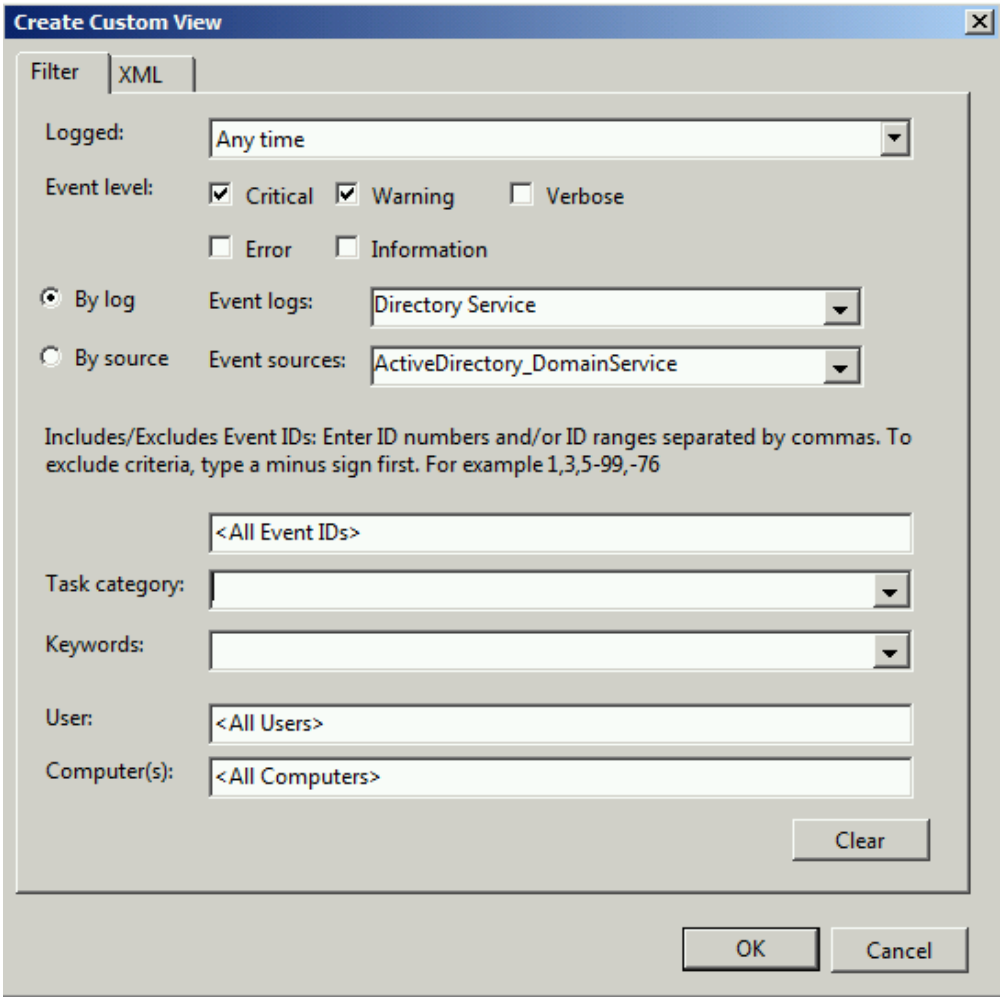


Mit dem Eintrag „Save Filtered Log File as“ können Sie nur die gefilterten Ereignisse in eine neue evtx-Datei speichern:

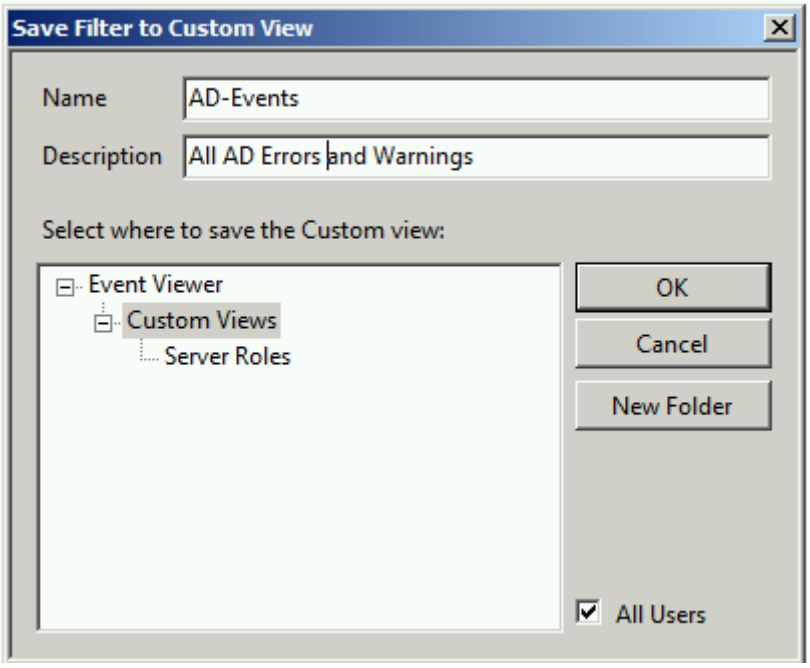


Und mit dem Eintrag Open Saved Log können Sie dieses Eventlog dann auch wieder mit dem Eventlog-Viewer öffnen.

Wenn Sie nach bestimmten Ereignissen regelmässig suchen müssen, lohnt es sich vielleicht, einen permanenten Filter zu definieren. Hierfür steht ihnen den Eintrag „Custom Views“ zur Verfügung. Hier können Sie im Kontextmenü „Create Custom View“ auswählen, um einen eigenen Filter zu definieren.



Hier definieren Sie, was der Filter anzeigen soll. Anders als bei einem normalen Filter wird dieser allerdings als XML-Datei gespeichert.



Mithilfe einer manuell erstellen XML-Abfrage können Sie sogar noch deutlich feiner filtern. Und keine Angst, die Abfragen sind sehr simpel. Folgende Query zeigt z.B. alle Anmeldung an, die über den RDP-Dienst (Remote Desktop, also mit dem Terminal Server Client mstsc) durchgeführt wurden.

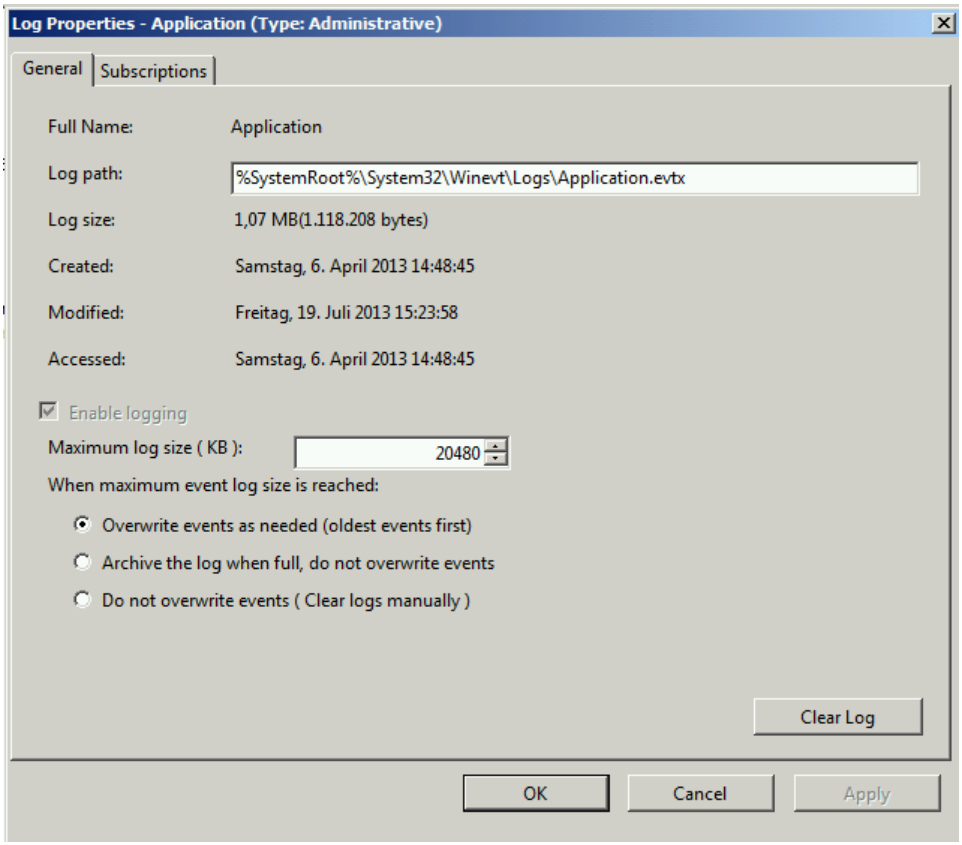
```
<Query Id="0" Path="Security">
  <Select Path="Security">
              *[EventData[Data[@Name='LogonType'] and (Data='10')]]
            </Select>
</Query>
```

Eine genaue Beschreibung finden Sie hier:

http://blogs.technet.com/b/askds/archive/2011/09/26/advanced-xml-filtering-in-the-windows-event-viewer.aspx

Eventlogs konfigurieren und pflegen

Eventlogs können sehr groß werden, obwohl das binäre EVTX-Format von Microsoft recht effizient arbeitet. Um trotzdem zu verhindern, dass ein Eventlog die Festplatte bis zum Bersten füllt, sind für Eventlogs maximale Größen festgelegt. Diese werden in den Properties (Ebenfalls im Kontextmenü des Logs) konfiguriert:



Die Maxmium Log Size definiert die maximale Größe des Files. Ist diese erreicht, kann man festlegen, wie Windows weiter mit dem Logfile umgeht:

Overwrite events as needed	Ist das Log voll, werden alte Ereignisse zuerst überschrieben
Archive the log when full, do not overwrite events	Das Eventlog wird mit einem Zeitstempel im Namen nach %SystemRoot%\System32\Config\ archiviert, das Eventlog wird geleert und ein neues Eventlog wird angelegt
Do not overwrite events	Es werden keine weiteren Einträge geschrieben und beim Anmelden wird eine Warnung ausgegeben, dass das Log voll ist.

Grundsätzlich ist die Option „do not overwrite events“ kaum zu empfehlen. Erwartet man eine große Zahl von Events im Log, so ist es sinnvoll, die Eventloggröße zu erhöhen (mehr als 200-300 MB machen keinen Sinn, da der Eventlogviewer dann sehr träge wird), und die Archivierung zu aktivieren. Allerdings muß man dann darauf achten, dass das Systemlaufwerk dann nicht irgendwann mit Eventlogs überläuft.

Eine ausführliche Beschreibung des Security Event Logs mit einer eingehenden Erläuterung der allgemeinen Einstellungen finden man hier (in Englisch):

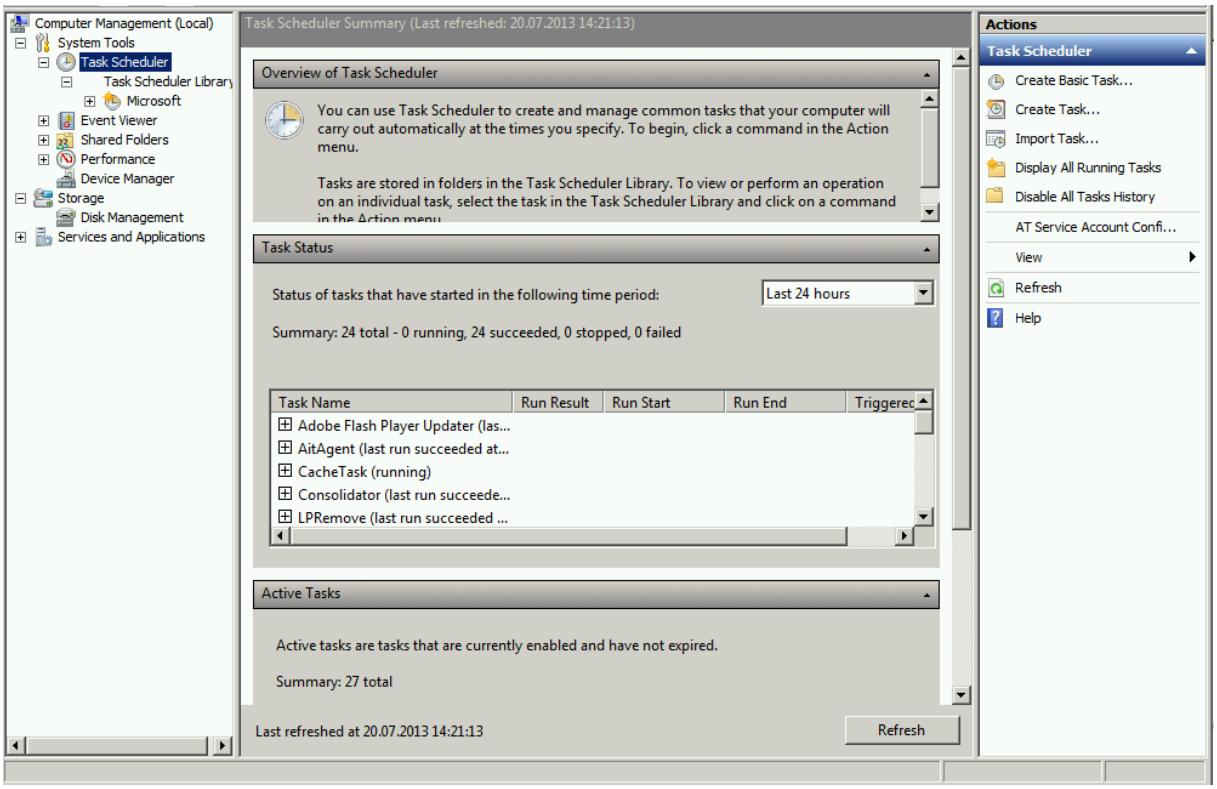
The Windows Server 2008 Security Log Revealed - Audit Policies and Event Viewer
http://www.ultimatewindowssecurity.com/securitylog/resourcekits/book2008/chapter2.aspx

Event-Logs per Gruppenrichtlinie konfigurieren
http://blogs.technet.com/b/askds/archive/2008/08/12/event-logging-policy-settings-in-windows-server-2008-and-vista.aspx

Geplante Aufgaben

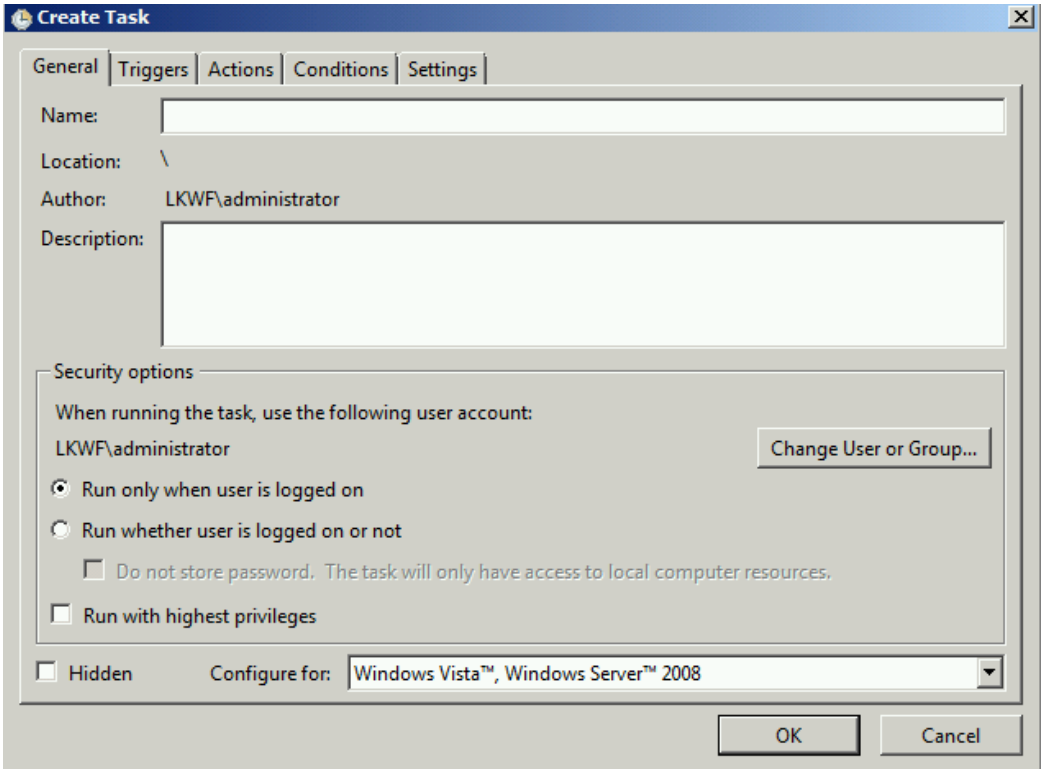
Um Aufgaben automatisch ereignisgesteuert – also zu bestimmten Zeiten oder aufgrund bestimmter Ereignisse – zu starten, gibt es bei Windows den Task Scheduler, oder in Deutsch die geplanten Aufgaben. Auch die Aufgabenplanung kann über die Computerverwaltung (compmgmt.msc) oder die Konsole „Task Scheduler“ gestartet werden – Scheduler kann man übrigens „Schedjuler“ oder „Skedjuler“ aussprechen, beides ist erlaubt, zumindest laut einem englischen Kollegen.

In der Ansicht der Aufgabenplanung findet man zuerst eine Übersicht mit aktiven Aufgaben und dem Status aktuell laufender Aufgaben. Außerdem findet man einen Unterknoten „Task Scheduler Library“.



Einen neuen Task legt man über das Kontextmenü des obersten Knotens Task Scheduler an, indem man *Create Task* auswählt. Create Basic Task ruft ein abgespecktes Menü auf, das nicht alle Optionen zur Verfügung stellt.

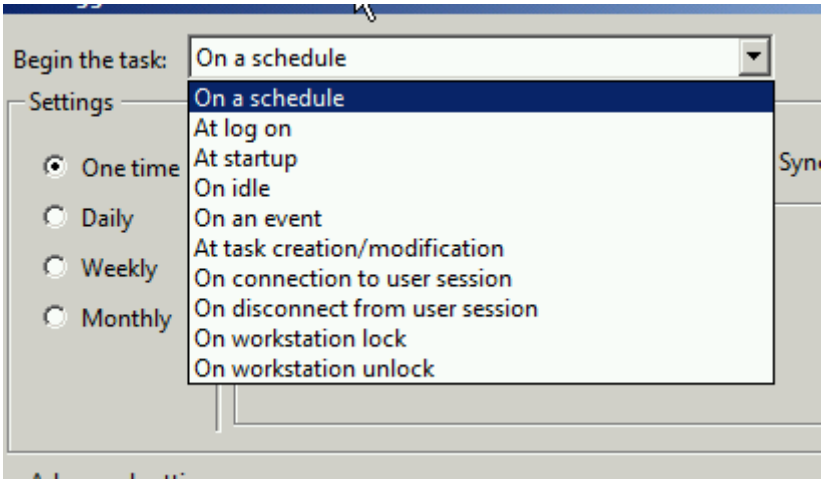
In einem Task können eine ganze Reihe von Daten hinterlegt werden. Diese sind über mehrere Reiter aufgeteilt:



General	Name, Beschreibung, und ganz wichtig das Benutzerkonto, unter dem die Aufgabe gestartet wird
Triggers	Definiert, wann der Task gestartet wird (=Auslöser)
Actions	Eine oder mehrere Aufgaben, die gestartet werden sollen. Möglich sind: • Ein Programm starten • Eine mail verschicken • Eine Meldung auf dem Bildschirm ausgeben
Conditions	Ausnahmen, unter denen der Job nicht gestartet wird – z.B. wenn der Rechner CPU-Last hat oder wenn er nicht am Stromnetz hängt (Batteriebetrieb)
Settings	Weitere Einstellungen wie z.B. wie lange die Aufgabe laufen darf

Trigger / Auslöser für geplante Aufgaben

Windows stellt eine ganze Reihe von Auslösern zur Verfügung, um geplante Aufgaben zu starten. Dies sind im Einzelnen:

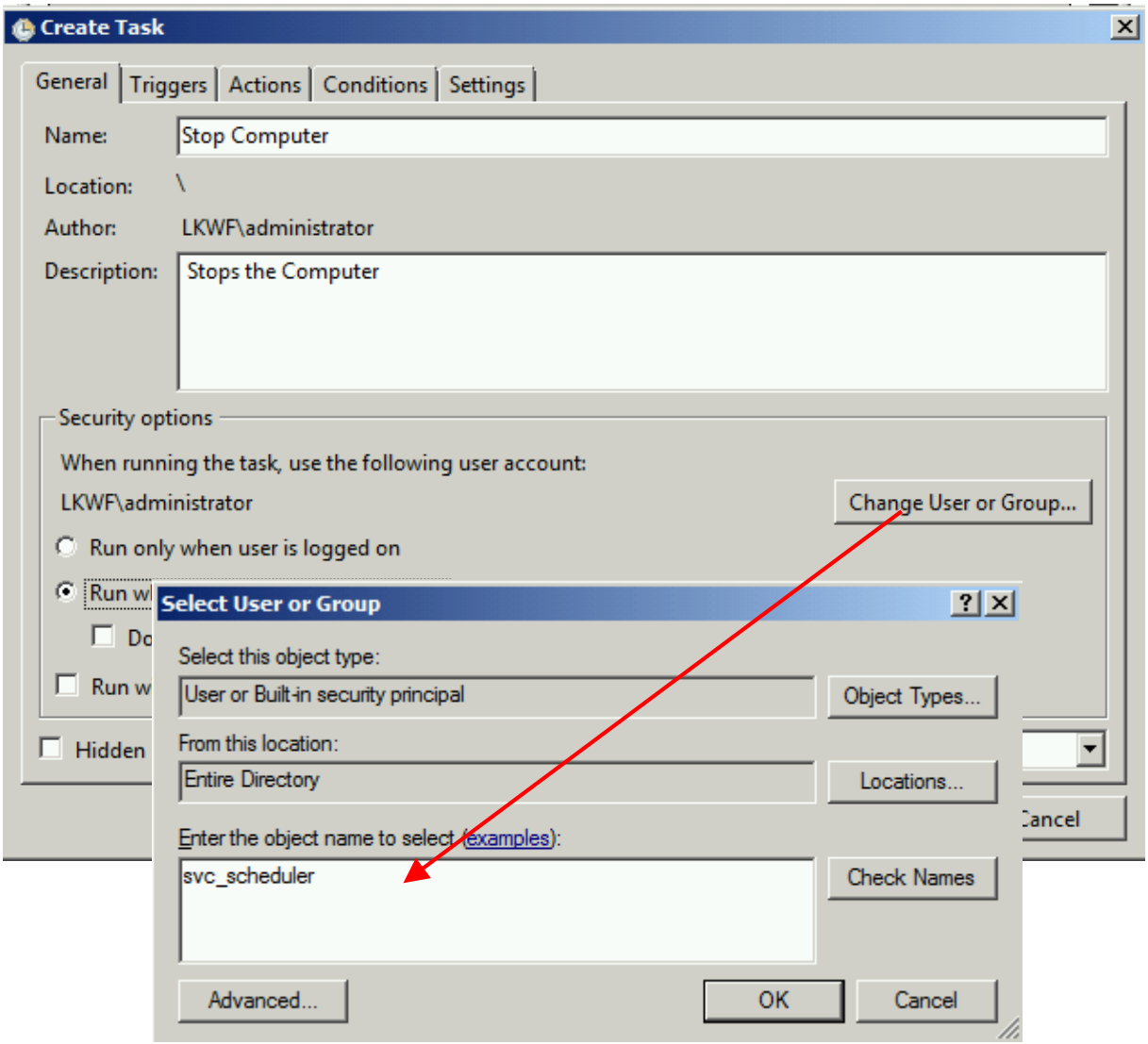


Auslöser	Beschreibung
On a Schedule	Zeitgesteuerter Start
At log on	Beim Anmelden eines Benutzers
At startup	Beim Starten des Computers
On idle	Wenn der Computer nicht verwendet wird. Wann das der Fall ist, hängt vom Betriebssystem ab. Eine genauere Definition finden Sie im Beispiel unten.
On an event	Wenn ein bestimmtes Ereignis im Eventlog protokolliert wird.
At task creation /modification	Beim Anlegen eines neuen Tasks
On connection to user session	Beim Anmelden per RDP (Remote Desktop Protocol)
On disconnect from user session	Beim Abmelden per RDP
On workstation lock	Wenn ein Benutzer den Rechner sperrt (z.B. mit Windows-Taste + L)
On workstation unlock	Wenn der Benutzer eine Sperrung aufhebt

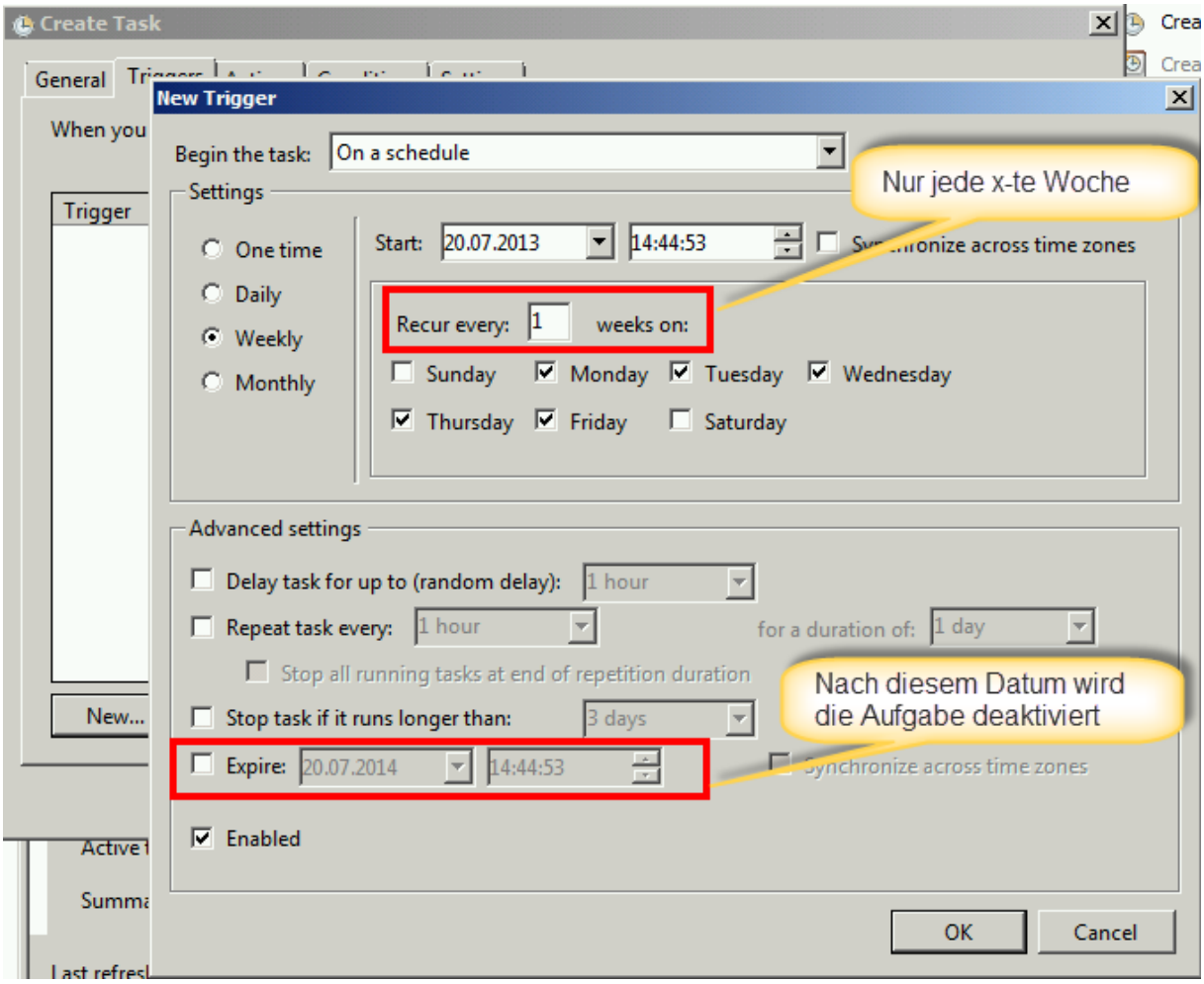
Diese Auslöser können aber noch einmal über die Einstellungen unter „Conditions“ beeinflusst werden.

Erstellen eine Jobs zum Herunterfahren des Computers

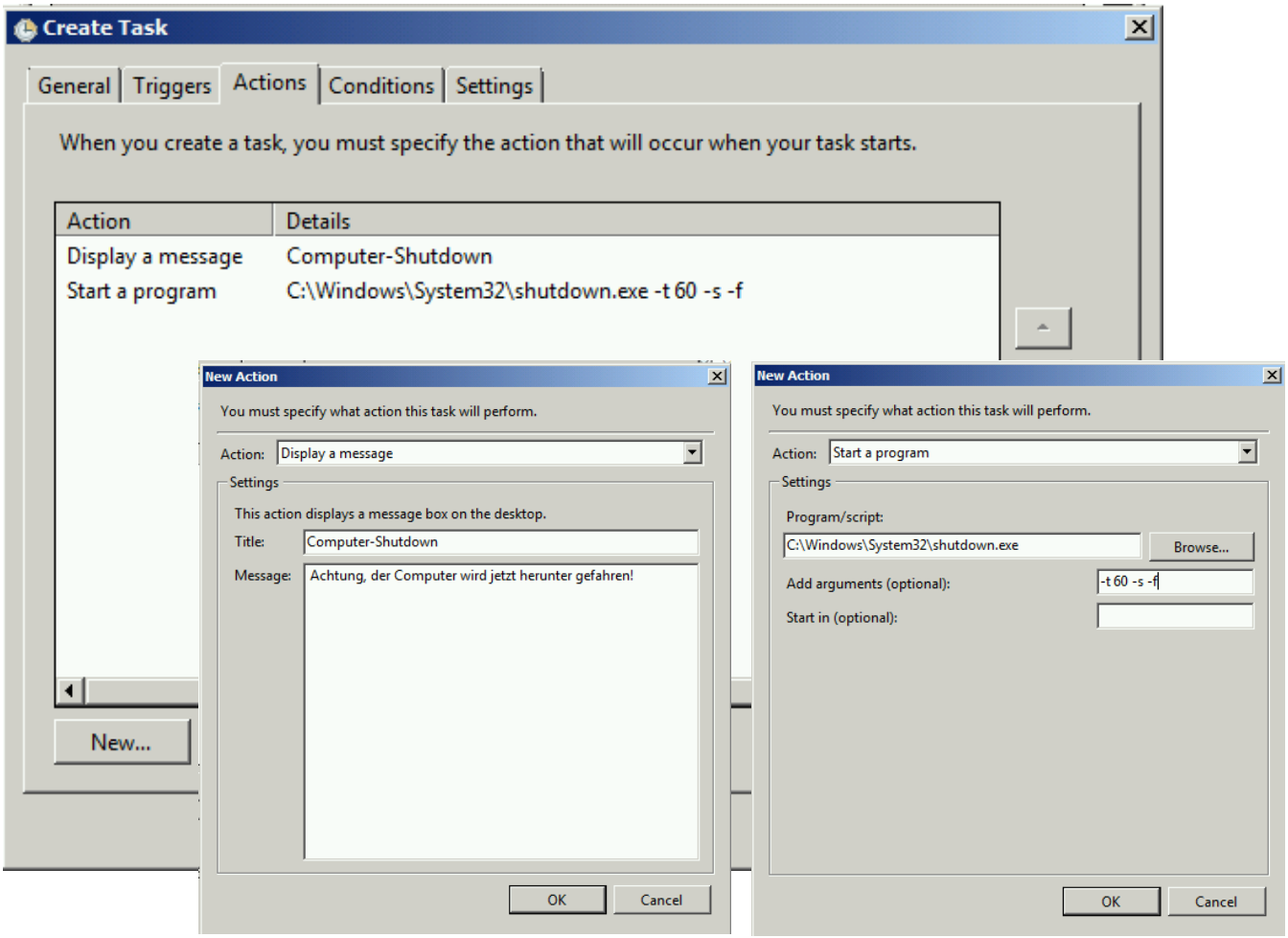
Am Beispiel wird eine geplante Aufgabe angelegt, die den Computer zeitgesteuert um 22 Uhr herunter fährt.



Es wird ein neuer Task erstellt und über „change User or Group“ als Benutzerkonto ein Servicekonto eingegeben. Das ist nicht zwingend notwendig, es könnte auch das vorgegeben Konto ausgewählt werden, oder die Aufgabe könnte mit dem Benutzer ausgeführt werden, der gerade angemeldet ist. Da unsere Aufgabe den Rechner auch dann herunterfahren soll, wenn kein Benutzer am PC angemeldet ist, müssen wir ein Konto hinterlegen. Das Kennwort des Benutzers wird beim Klicken auf OK abgefragt und sicher gespeichert.

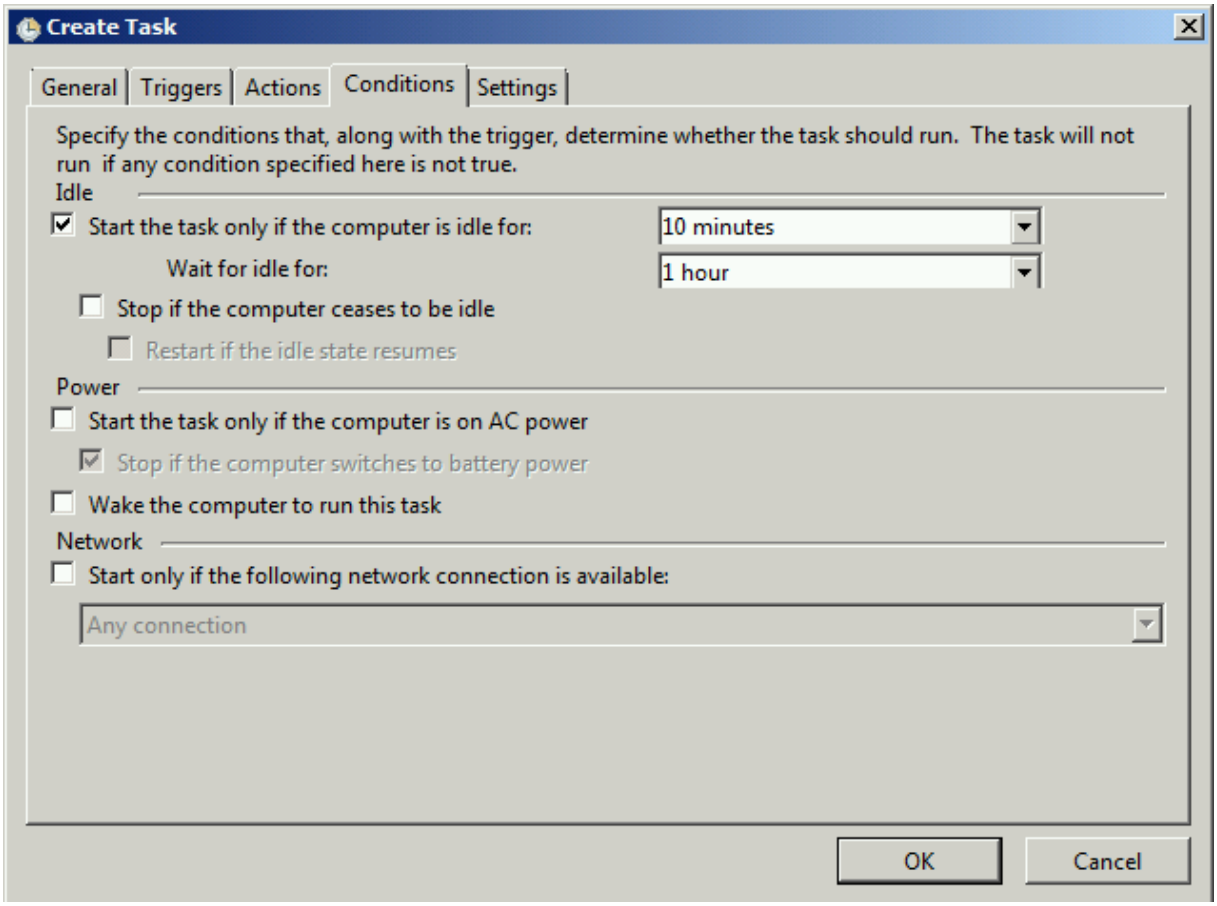


Wir legen unter dem Reiter Trigger einen neuen Auslöser für die Aufgabe an. Da der Task zeitgesteuert gestartet werden soll, wählen wir *on a schedule* aus und wählen „Weekly“, da wir hier die einzelnen Wochentage wählen können, an denen die Aufgabe laufen soll. Hier könnten wir auch festlegen, dass die Aufgabe zeitverzögert starten soll, falls mehrere Aufgaben zur gleichen Zeit starten und man diese entzerren möchte. Der Zeitversatz ist dann zufällig. Außerdem kann die Aufgabe mehrfach hintereinander ausgeführt werden oder automatisch beendet werden, falls die Aufgabe zu lange läuft.

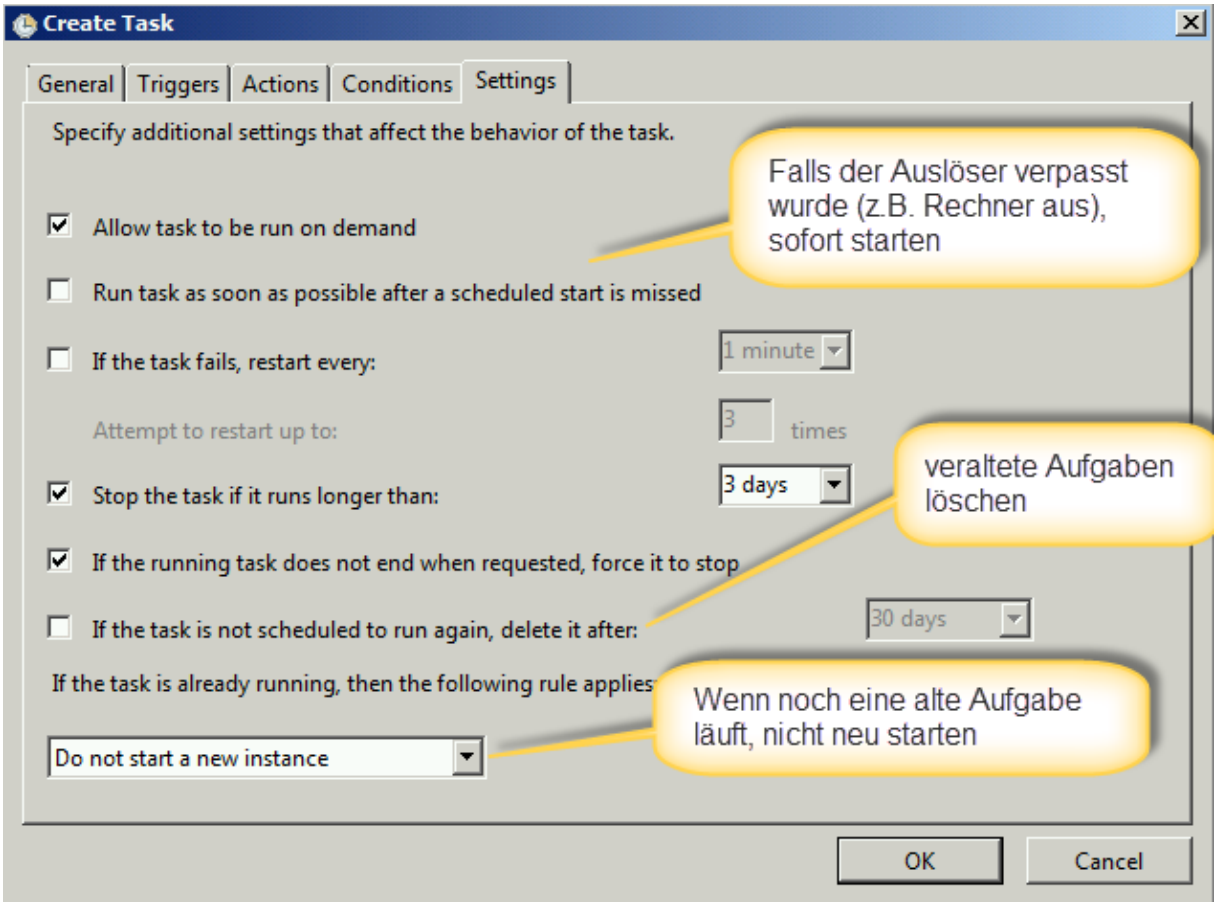


Nun definieren wir unter Action 2 neue Aktionen, indem wir auf *New* klicken und zuerst Aktion 1 erstellen – *Display a message-,* dann mit OK bestätigen und mit *New* eine weitere Aktion – *Start a program* – hinzufügen. Wir wählen aus %systemroot%\system32 die shutdown.exe aus, und geben als zusätzliche Argumente an:

-t 60	60 Sekunden mit dem Herunterfahren warten
-s	Den Computer herunterfahren. Mit –r erzwingt man einen Neustart
-f	Force, erzwingt das beenden von Programmen, die z.B. aufgrund einer „Speichern“-Anfrage nicht auf die Schließen-Anforderung des Betriebssystems reagieren.

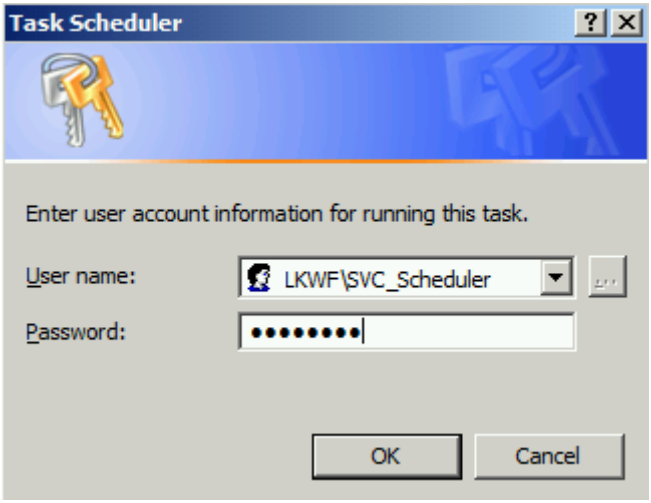


Nun definieren wir noch, dass der Computer nur dann herunter gefahren wird, wenn er idle ist, also keine CPU-Last hat und kein Benutzer angemeldet ist. Die Idle-Bedingungen sind abhängig vom Betriebssystem. Für Windows 7 gilt beispielsweise, dass 15 Minuten lang keine Benutzereingaben mit Maus und Tastatur gemacht werden durften, und dass in den 15 Minuten die CPU und die Festplatte zu 90% ungenutzt waren. (Siehe hierzu: http://msdn.microsoft.com/en-us/library/windows/desktop/aa383561%28v=vs.85%29.aspx). Durch *wait for idle* wartet Windows bis zu 1 Stunde auf diese Bedingung, bevor es die Ausführung verwirft. Außerdem haben wir die Bedingung, dass der Computer nur unter Netzstrom die Aufgabe ausführt, deaktiviert.



Unter dem Reiter Settings können wir festlegen, dass der Task auch manuell (on demand) gestartet werden kann. Interessant ist hier noch die Option, die geplante Aufgabe automatisch nach einem Zeitfenster zu löschen, wenn sie eine gewisse Zeit nicht genutzt wurde. Weiterhin kann festgelegt werden, dass eine Aufgabe immer nur einmal gestartet wird, was z.B. spannend ist, wenn die Aufgabe noch läuft, aber der Zeitpunkt für einen weiteren Start schon erreicht ist.

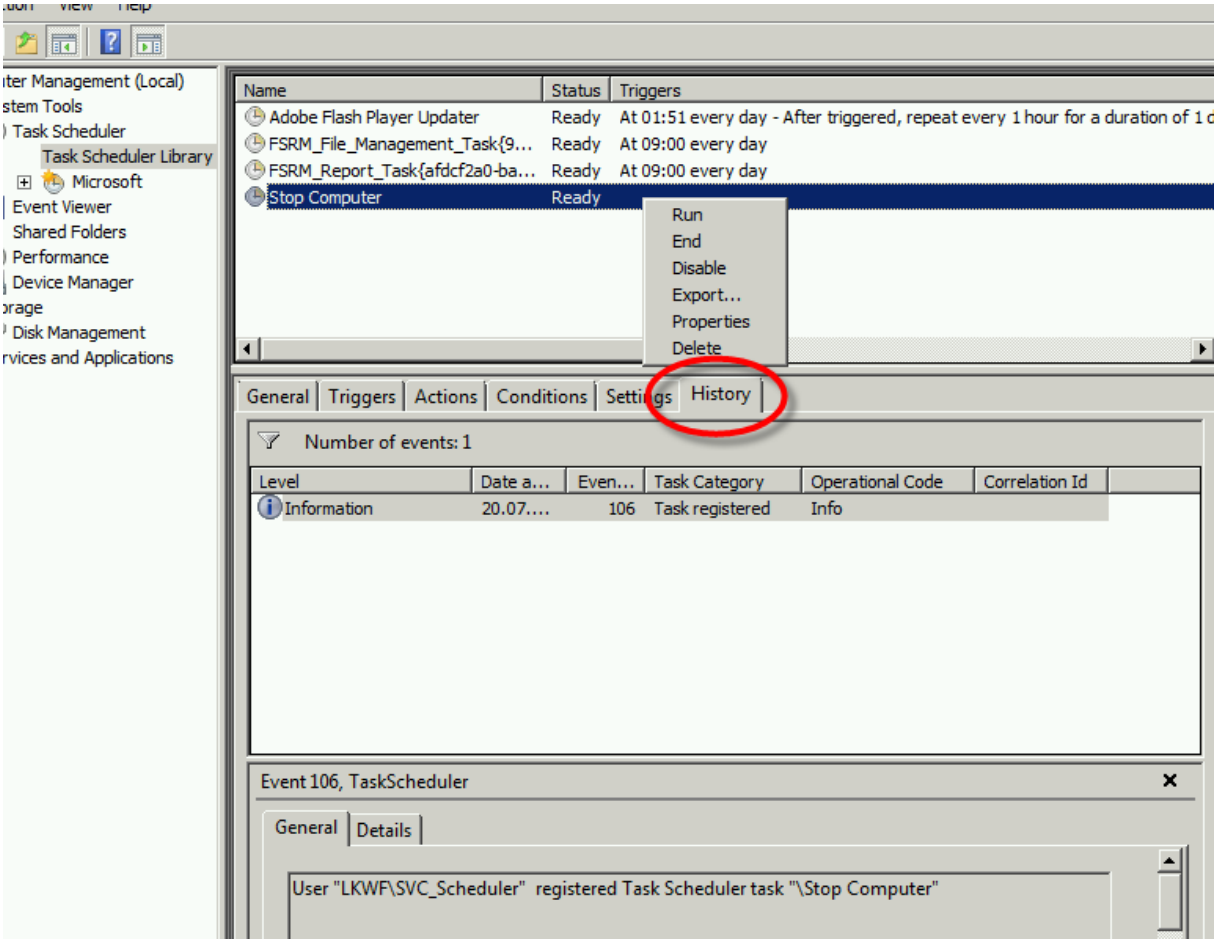
Mit Klick auf OK wird das Kennwort des Benutzers abgefragt, mit dem der Task gestartet werden soll:



Sollten Sie hier eine kryptische Fehlermeldung bekommen, prüfen Sie bitte, ob der Benutzer administrative Rechte hat. Diese werden für den Rechnerneustart benötigt.

Die geplante Aufgabe finden Sie jetzt, wenn Sie in der Verwaltungskonsole die *Task Scheduler Library* auswählen. Hier können Sie alle von Ihnen definierten Aufgaben sehen, deren Einstellungen und im

Reiter History den Verlauf inkl. Fehlermeldungen. Wenn Sie also wissen möchten, ob Ihre Aufgabe korrekt gelaufen ist oder falls ein Fehler aufgetreten ist, welcher das war, finden Sie hier alle Informationen.



Außerdem können Sie hier über das Kontextmenü den Job manuell starten, soweit Sie das bei der Definition zugelassen haben, oder den Job deaktivieren. Auch sehr interessant ist das Exportieren. Da Windows keine Möglichkeit der zentralen Steuerung von Jobs gibt, können Sie hier zumindest die Definition eines Jobs in ein File exportieren und auf anderen Rechnern den Job in seiner kompletten Konfiguration wieder übernehmen, ohne den Job jedes Mal komplett neu definieren zu müssen.

Eine Beschreibung bei WINFAQ:
http://www.winfaq.de/faq_html/Content/tip2500/onlinefaq.php?h=tip2548.htm

Windows Powershell

Die Windows Powershell ist eine Kommandokonsole, die das automatisieren von Windows-Systemen deutlich erleichtern soll. Einen kleinen Kurs für Windows Powershell 3.0 in Deutsch finden Sie hier:

http://bookboon.com/de/windows-powershell-3-0-ebook

Einige interessante Commandlets:

`Get-command –Type cmdlet`	Zeigt alle Commandlets (Powershell-Befehle) an
`Get-eventlog –Logname Application –Message *log*`	Zeigt aus dem Eventlog alle Einträge, die in der Beschreibung log stehen haben
`Get-Eventlog –Logname Application –Message *log* -EntryType Error`	Wie oben, aber nur Fehlermeldungen
`Get-Eventlog –Logname System | Export-CSV –NoTypeInformation C:\temp\Systemlog.csv`	Gibt das System-Eventlog als CSV-Datei (Komma-separierte Textdatei) in C:\temp\systemlog.csv aus
`Get-help`	Zeigt die Powershell-Hilfe an
`Get-service`	Zeigt alle laufenden Dienste an
`Get-service | format-list *`	Zeigt alle Dienste mit allen Ihren Eigenschaften in Listenform statt tabellarisch an
`Get-process`	Zeit alle laufenden Prozesse an
`Get-process | select-object –first 1`	Zeigt nur den ersten Prozess an, denn get-process ausgibt