Wieso gätn des net…

Günter Hartl

23 Uhr 48, Dein trüber Blick wandert resignierend über das auf deinem Bildschirm ausgebreitete Szenario. Eine dicke fette Dropbox mit dem unmissverständlichen Auftrag, Dich über fehlende Zugriffsrechte zu informieren.

Weder stundenlanges „*try and error*" noch Panikanrufe bei Freunden brachten bisher ein befriedigendes Ergebnis?

Willkommen im Klub…

Hier kommen wir zu einem Thema, mit dem der Privat-User bisher nur rudimentär, wenn überhaupt konfrontiert wurde.

Der Rechteverwaltung in einem Betriebssystem

Vor allem in der Windows-Welt sorgt dieses Thema immer wieder für Zündstoff. Völlig zu Unrecht, wie ich meine. Ich gebe schon zu, dass diese Thematik am Anfang recht furchteinflößend und verwirrend wirken kann. Hat sie anfangs auf mich auch.

Aber man hat immer nur Angst vor dem, was man nicht kennt.

Legende

Token: Eine Komponente zum Identifizieren des jeweiligen Benutzers

UAC: User Access Control > Benutzerkontensteuerung Vista+Windows 7

Administrator: Kann sich alle Rechte auf einem Windows-System verschaffen

NTFS-Rechte: damit kann ich den Zugriff auf jede Datei in Windows regeln

Freigaberechte: damit regle ich den Zugriff auf Ressourcen im Netzwerk

Windows-NT: NT-4,W2k, XP,Vista,7 > Merkmal „Mehrbenutzersystem"

Windows-wos-was-i: 95,98,Me > typische Einzelplatzsysteme

Verzeichnistrenner: / bei Linux, \ bei Windows

Dateinamen: Groß-und Kleinschreibung in Linux beachten, in Windows egal

Root: Superuser in Linux = wie Administrator in Windows

Home-Verzeichnis: Heimatordner für jeden User auf einem Linuxsystem

Root-Ordner: Heimatordner für root.

/: Wurzelverzeichnis = oberster Ordner im Linux-Dateibaum

Rechteverwaltung: Kann ich in Windows deaktivieren, in Linux nicht.

Rechteverwaltung: in Windows granular einstellbar (UAC), in Linux nicht

Userverwaltung: Jeder User braucht ein Passwort in Linux, in Windows nicht

Root: grafische Anmeldung nicht vorgesehen > Sicherheitsrisiko

Administrator: Grafische Anmeldung in Windows möglich

Als „root" musst Du in Linux wissen, was Du tust… ganz wichtig

So, das wär's im Großen und Ganzen.

Wenn'st jetzt noch weiterliest, erklär ich das unten genauer; das wird wohl ein längerer Artikel… Richtig. Diese Zeilen entstanden über einen Zeitraum von mehreren Monaten, da durch die tägliche Praxis immer wieder neue Erkenntnisse und Erfahrungswerte eingeflochten werden mussten.

Des weiteren sind einige Passagen sehr technikklastig ausgelegt, andere wiederum ausschließlich der Praxis gewidmet. Um aber in der Praxis nachvollziehbare Operationen zu begreifen, kann eben ein technisches Hintergrundwissen zumindest nicht schaden. Im Laufe des Artikels wirst Du auch sehen, dass die Rechteverwaltung kein wirklich triviales Thema ist. Selbst für Profis nicht.

Du musst also nicht alles auf einmal lesen. Wird sonst zu anstrengend.

Die Verbreitung der „neuen" Windows-Betriebssysteme (Vista + Windows 7) konfrontiert die meisten Anwender mit etwas, was es in dieser transparenten Form noch nie vorher gegeben hatte. (obwohl es immer schon da war…)

Der Rechteverwaltung in einem Windows-NT-Betriebssystem

Die Intention der nachfolgenden Zeilen hat trotz der breitgefächerten Betrachtungsweisen nur ein Ziel: Verständnisfragen der Rechteverwaltung zu beantworten.

Im Laufe der Zeit bin ich draufgekommen (und viele andere wahrscheinlich auch), dass das Aufzeigen der Beweggründe für eine Lösung, gleich welcher Natur, einer fertigen Endaussage immer vorzuziehen ist.

Und ganz wichtig: die Anwendung der sokratischen Differenz. Heißt, man muss es dem Gegenüber so erklären, dass es dieser auch versteht.

Wir alle kennen Techniker/innen *(gut gedschändat, so viel Zeit muss sein)*, die sich schon nach 2 Sätzen hinter ihren Fachtermini verschanzen und Dich dann volllabern. Hier muss ich sozusagen die literarische Blutgrätsche vollziehen, um den absoluten PC-Anfänger genauso wie den erfahrenen Windows-Admin mit meinen Zeilen anzusprechen.

Jetzt wieder mein Standardsatz: Ja, es gibt 1000 Wege. Ich nehm' aber immer die „Trottelversion". Die hat den entscheidenden Vorteil, dass sie funktioniert. Auch wenn es elegantere Lösungen geben sollte. Es sollte für jeden Heimuser nachvollziehbar sein. Eine praxistaugliche Hilfestellung eben.

Was ich damit sagen will: Nach dem Durchlesen dieses Artikels solltest Du das Rechtesystem in Windows und Linux nachvollziehen können und eventuelle Problemstellungen logisch lösen können.

Des Weiteren befasst sich der Artikel zusätzlich noch mit den konzeptionellen Unterschieden der Rechteverwaltung zwischen Windows und Linux.

Ich verspreche Dir, das sieht am Anfang alles irrsinnig kompliziert aus. Ist es auch. Vor allem, wenn man keine Ahnung hat. Wenn Du aber nach dem Durchlesen diese Artikels an Deiner Kiste die Berechtigungen durchgehst, einen neuen Account anlegen solltest oder einfach nur was kopierst von einem externen Datenträger, werden Dir eine Menge „aha"-Erlebnisse widerfahren. Versprochen.

Dieser Artikel soll (und kann) auch kein Ersatz für die Aneignung tiefgehender Kenntnisse in der User- und Rechteverwaltung sein. Weder für Linux noch für Windows. Folglich habe ich exemplarisch „nur" ein paar der auffälligsten (oder üblichen) Szenarien von „Problemen" angerissen, die einem Heim-User widerfahren könnten. Es sei noch einmal explizit darauf hingewiesen, dass meine Zeilen bei Weitem kein vollständiges Curriculum der Rechteverwaltung darstellen (können).

Davon abgesehen habe ich bewusst auf eine strikte Gliederung zwischen der Linux und der Windows-Rechteverwaltung verzichtet, einfach weil ich jetzt einmal wild drauf losschreibe… und mein Chefredakteur (Fiala) wird das wie üblich in die richtigen Bahnen lenken.

Noch was in eigener Sache: Ja, auch mich traf es wie ein Keulenhieb. Das Outing meines Chefredakteurs bezüglich seiner tschechischen Wurzeln fegte in der letzten PCNEWS-Ausgabe wie ein Tornado durch die aufgewühlte, verstörte Leserschaft.

Eine ehrenwerte Computerzeitschrift als Plattform für seinen bisher verheimlichten (es gilt die Unschuldsvermutung) Migrationshintergrund zu entfremden, äußerte sich in einer gewissen Distanzierung (und unendlichen Traurigkeit) meinerseits.

Nach ein paar fruchtlosen Versuchen meinerseits, ihn zumindest verbal an unserer Gesellschaft teilhaben zu lassen à la „…Franz, da Du Knopf drücken musst, dann kommen Aufzug…" „…geh, was i äh, trotzdem danke…", betrachtete ich meinen selbstlosen Integrationsauftrag als erfolgreich beendet.

Um die inzwischen ausgezeichneten bilateralen Beziehungen zu unterstreichen, werde ich Herrn Fiala deshalb beim nächsten Club-Treffen medienwirksam in meine Keksdose greifen lassen…

Let's go… wie der Lateiner sagt.

Rechteverwaltung? Wer braucht schon so was? Jede Firma zum Beispiel. Die lassen wir aber großteils in diesem Artikel außen vor. Wir widmen uns hier hauptsächlich der Arbeitsumgebung eines Privatusers. Der braucht natürlich auch eine Rechteverwaltung, die seine privaten Daten zum Beispiel vor Fremdzugriffen schützt.

Jedes performante Rechtesystem auf einem Betriebssystem, egal ob Linux oder Windows, bedingt ein entsprechendes Dateisystem *(filesystem)*.

Was bisher in der Rechtefrage geschah

Unter Windows98 waren wir alle doch recht glücklich mit FAT32. *(hüstel.)*

Aber schon damals (1993 gab es das NTFS-Dateisystem für Windows-NT)

NTFS= *New Technology File System*

Jetzt kommt der vielleicht wichtigste Satz des ganzen Artikels.

Die Rechteverwaltung hat sich seit Windows-XP grundlegend verändert. Sollte man vorher

vielleicht zur Kenntnis nehmen, bevor man auf Windows 7 (oder Vista) umsteigt.

Lies den Satz noch einmal. Verinnerliche ihn Dir gut. Er hilft ungemein bei eventuellen Problemlösungen in der Rechteverwaltung.

Die Rechte- und Userverwaltung hat sich in der Linux-Welt seit 20 Jahren (für den Privatuser) nicht geändert.

Sorry, für geübte Linux-User wird nicht viel Neues dabei sein.

Diese Aussage mag auf den ersten Blick sehr provokant wirken. Soll sie auch…sonst döst Du mir noch ganz weg.

Ich füge noch ein paar Ergänzungen hinzu.

Wenn Du das Rechtesystem in Windows7 nicht verstehst, keine Angst, die meisten Privatuser haben auch keinen Tau davon. *(„wieso isn des so kompliziert…do kann I ja glei Linux nehman…")*

Eigentlich hätten die Windows-User schon bei Windows2000 bezüglich der Rechteverwaltung schreien müssen. Oder zumindest ein krakelender Mob die Reaktivierung von Windows98 (das ja noch keine ausgereifte Rechteverwaltung zu bieten hatte) lautstark einfordern können…

vor Windows95/98/ME = Einzelplatzsystem > keine Rechteverwaltung

nach Windows2000 = NT-Linie > Mehrbenutzersystem > Rechteverwaltung

Windows2000 war das erste Betriebssystem der NT-Linie von Microsoft, das einer breiten Öffentlichkeit zugänglich gemacht wurde. Ein Merkmal dieser Version war schon damals die NTFS-Rechteverwaltung.

W2k (Windows 2000) löste die bis dahin vorherrschenden Betriebssysteme 98 und ME im Privatbereich ab. Die Geburt der NT-Linie für den breiten Massenmarkt war geboren.

(NT3-4 war eher im professionellen Bereich beheimatet).

Ein großer Mehrwert von W2k war wie schon erwähnt die strikte Rechteverwaltung, die aber im privaten Umfeld selten angewendet wurde.

Darum regte sich damals auch keiner über das einschränkende Rechtesystem auf.

Es war auch das letzte NT-System, das es „nur" in einer Version gab. (Okay, Server 2000 gab's noch, aber eher uninteressant für den Heim-User).

Daraus ließ sich auch der gute Ruf von Windows2000 ableiten, weil eben nur die höchstwertige Version (Professionell) zu erwerben war. Das Beste eben auch für den Heim-User.

XP hingegen war das erste Microsoft-NT-Produkt, das sowohl für den Heimanwender, als auch für das Unternehmensumfeld konzipiert wurde. Man ging einfach dazu über, verschiedene Versionen von XP (abgestuft nach deren Features und Verwendbarkeit… *starter-, home-professional*…) auf den Markt zu werfen.

Konträr zu Windows 2000 wurde mit den XP-Home-Versionen explizit der Heimanwenderbereich abgedeckt. Eigene Home-Versionen gab es bei Windows 2000 ja (noch) nicht.

Einer der Hauptgründe, warum auch heute noch teilweise Windows2000-Boxen in Unternehmen laufen.

Die Home-Versionen (XP, Vista, 7) konnten (und können) auch heute nicht in großen Unternehmen eingesetzt werden, da sie nur eingeschränkte Features haben.

Vor allem bei den Netzwerkfähigkeiten, User- und Rechteverwaltung und der Verschlüsselung.

Defaultmäßig war man im Heimbereich sowieso als Administrator angemeldet, das mit einem Außerkraftsetzen der Rechteverwaltung gleichzusetzen war.

Im Privatbereich interessierte das eh keinen. *(„i bin Admin, do derf i ollas mochn.")*.

Mit dem Erscheinen von Vista wurde seitens von Microsoft ein Paradigmenwechsel (sorry, hab das Wort auch erst kürzlich vom Duden ausgeborgt…) eingeleitet, der dem User eine neue Rolle im System zukommen lässt. Nämlich die eines Benutzers mit defaultmäßig eingeschränkten Rechten. Das geht schon mal zumindest in Richtung Linux. Endlich…

Microsoft musste daher etwas unternehmen, um seine User in diese Richtung zu sensibilisieren. Das augenfälligste Instrument dafür war (und ist) die UAC.

Da diese eine strikte Rechteverwaltung nach sich zieht (oder zumindest dafür angedacht war), häufen sich natürlich die „Beschwerden" bezüglich fehlender Rechte *(„in XP hot des no funktioniert.wieso gät des net.")*.

Problematisch wird's meist, wenn durch einen Umstand (Festplattentausch, Upgrade, externe Datenträger, Systemfiles…) Berechtigungsprobleme auftreten.

Die Probleme „entstehen" deshalb, weil jetzt zum ersten Mal in der Geschichte von Microsoft ein Rechtesystem in Windows (Vista + Windows 7) standardmäßig bei jedem User zum Tragen kommt.

Wie erwähnt, war das Rechtesystem in Windows2000 im Privatbereich eher ein optionaler Faktor. Oder besser gesagt, wurde das Rechtesystem durch den Benutzer, der zugleich Administrator war, ausgehebelt.

Sprich, es gab keine Einschränkungen für den User. XP ditto. Das gute alte Windows 98-Feeling wurde man so schnell nicht los…

Die Microsoft-Masche funktioniert seit Jahren nach diesem Prinzip. Spätfolgen sind daher nur mit einem extremen Kraftaufwand beizukommen.Wenn überhaupt.

(Einführung der Registry zugunsten der SAM, NTFS statt Fat32, Multiuser statt Einzelplatzsystem, UAC statt automatischem Administratoraccount, WAT+WGA , im Betriebssystem verankerter IE statt freier Browserwahl.... usw.).

Es spricht ja nichts dagegen, dass man Geld mit dem Zeug verdienen soll. Aber bitte, dann machts das gleich richtig und stückelts nicht immer was dazu. Alleine die Registry-Philosophie und das längst versprochene und überfällige Dateisystem könnten Bände füllen.

Es wurde von Microsoft praktisch „geduldet", dass der normale User (bis zu XP) automatisch als *„Administrator"* auf dem Pc unterwegs ist. Viele Heimuser wissen (oder wussten) ja bis zu Vista gar nicht, dass es Benutzerkonten mit eingeschränkten Rechten überhaupt gibt. Und vor allem, dass man die verwenden sollte. *(„wieso, i bin äh ois Karl angmeldet am System...")*.

Dank Microsoft darf (und kann) auch der erkenntnisresistenteste (und laienhafteste) User ein Betriebssystem installieren und warten. Na klar glauben das die Leute von der Straße noch heute.

Microsoft hats ja selbst gesagt. *(äääh...was sollens sont sagen, dass man Fachkenntnisse benötigt? Da würde die defragmentiererprobte Alleinerzieherin ja gleich im Media-Markt auf dem Absatz kehrt machen und zu Linux wechseln können.)*

Schick' einmal das EDV-Team in Deiner Firma für einen Tag nach Hause… und dann schau einmal, was passiert...

Das bekommt man nur schwer aus den Köpfen raus:

Windows: kann jeder bedienen, installieren, reparieren und warten

Linux: Nur für Profis, kompliziert, muss man programmieren können

Die Umerziehung des Users ist der größte Brocken, den Microsoft zu bewältigen hat. Unbestritten.

Wer erinnert sich noch an die Anfangszeiten, als Microsoft ihre Betriebssysteme und Office-Pakete verschenkt hat? War Anfang der 90er Jahre glaube ich. Was hatte das für einen Sinn? Ganz einfach. Die Leute an das Zeug zu gewöhnen. Die Gratiskeule wirkt da immer sehr gut. Auch heute noch.

Nach der Gewöhnungsphase und Akzeptanz der Produkte wurde das Ganze kommerzialisiert. Sprich, Kohle mit Lizenzen eingefahren.

Das hast Du heute noch in abgeschwächter Form bei Office-Paketen, die sich nach 60 Tagen deaktivieren, wenn man sie nicht käuflich erwirbt. Virenscanner ditto.

Da die meisten User aber keinen Bock hatten *(i zoi nix, de kennan mi...)*, für Microsoft-Lizenzen Geld abzudrücken, die sie ja vorher geschenkt bekommen hatten, verkomplizierte sich der Kauf, die Installation und Wartung der Produkte mit den Jahren exponentiell.

Heute leben wir wie selbstverständlich mit *Windows Genuine Advantage* (WGA > bei XP) und *Windows Activation Technologies* (WAT > bei Vista und 7).

Diese Tools dienen ausschließlich als Stolpersteine für Raubkopien. Ehrlich, wie viele Deiner Freunde verwenden eine legale Lizenz von Microsoft-Produkten? Das hat sich über Jahre manifestiert, dass man für Microsoft-Produkte nichts zahlt (zahlen soll). Vor allem im Privatbereich und bei KMUs ist diese Philosophie sehr verbreitet. *(wieder > „bin i deppat....)*. Fehlende Datensicherung ditto.*(...ahh, wird scho nix sei...)*

Auf ein Wort: Jeder ist überzeugt, dass Datensicherung ein absolutes MUSS ist.

Dieser Satz kommt aber immer erst nach dem Supergau.

Genauso mit der (fehlenden) Rechteverwaltung. *„ ..brauch i net… zu kompliziert… bin e nur allan ich am PC..."*.

Wieder: die Umerziehung des Users ist die größte Hürde. Vom Einzelplatzsystem zum Mehrbenutzersystem ist's ein weiter Weg. Vor allem in den Köpfen der User. Microsoft half da nach, indem jetzt automatisch der User am System eingeschränkte Rechte hat.

Was man da Microsoft vielleicht vorwerfen kann: Es wurden immer nur die „schönen" Seiten der EDV angepriesen. Dadurch trat so was wie ein Gewohnheitseffekt ein, der praktischerweise gleich als „der PC-Standard" einbetoniert wurde. Jeder PC „muss" so wie Windows funktionieren.

Darum gab es früher auch sehr viele Umstellungsschwierigkeiten von Usern, die von XP auf Linux wechseln wollten... *„wieso hob i do kan Zugriff...wieso derf i des net...."*. Die waren natürlich ordentlich gefrustet und blieben erwartungsgemäß nicht lange bei Linux.

Deshalb wurde (und wird) Linux hauptsächlich als kompliziert und einschränkend empfunden (hauptsächlich von Windows-Powerusern). Mir sagen noch heute User, denen ich ein Linux

installiert habe, dass ich sie *„ganz schön eingeschränkt und ihnen alles abgedreht habe...“*. Wenn ich erwidere: *„...das war nicht ich, sondern das System...“*, glauben sie mir trotzdem nicht.

Dann kommt meine Killerfrage: *„Geht irgendwas nicht?“*.

Antwort: *„Nein, nein, geht e alles, aber is schon komisch, wo ich doch alleine am PC arbeite, dass ich nicht überall Zugriff habe“*.

Heute mit Windows7 hat man eine ähnliche Situation, da man jetzt auch nicht mehr überall im System herumfuhrwerken kann. Glaub‘s mir, das dient hauptsächlich der Systemsicherheit.

Was Linux betrifft, gab es solche Hindernisse im Gedankenmuster des Users nie. Linux war von Anfang an als Mehrbenutzersystem ausgelegt. (Mit all seinen Vor- und Nachteilen). Genauso wie Windows-NT Mitte der 90er Jahre, das aber nicht für den Privatuser angedacht war.

Aus diesem Grund war auch keine aufwändige Pionierarbeit (2007 bei Vista) mit UAC und dergleichen notwendig, damit die User ihr Betriebssystem richtig benützen.

In Linux arbeitest Du immer mit einer vollwertigen Rechteverwaltung. Auch auf einem Linux, das von einer 1,44 MB großen Floppy läuft, hast Du die selbe Rechteverwaltung wie auf einem Linux-Mailserver der Nationalbank.

Die Rechteverwaltung wurde auch nicht von Linux „erfunden“. Sondern ist ein Unix-Feature. gibt’s also schon seit den 70er Jahren. Apple hat das auch großteils übernommen.

Noch ein Tipp: Wenn Dir Freunde empfehlen, dass Du den UAC-Schieberegler in Windows ganz nach unten geben solltest… such‘ Dir neue Freunde oder welche die Ahnung von Windows haben.

Auf lange Sicht wirst Du mit dieser Herangehensweise nicht glücklich werden.

Gewöhn Dich an die Rechteverwaltung, ist nicht so schlimm, ehrlich.

Wenn demgegenüber heutzutage bei Windows7 irgendwas bei den Berechtigungen nicht gleich hinhaut, schlägt dann meist die alte Windows-Krankheit durch. Die manifestiert sich darin, dass ziellos Hakerln in aufpoppenden Dialogboxen platziert oder entfernt werden.

Nach stundenlangem Herumprobieren darf sich mit der Zeit die konsultierte Verwandtschaft *„mei Bua kennt si aus, der orbeit jedn Tog min Pc. da Klausi hot ma scho oft in PC gricht.äähh, PC richten heißt neu installieren in Windows :-)“* auch noch am PC wichtig machen.

Vielleicht hat man Glück und einer hat‘s wirklich hinbekommen. Die Regel ist es aber nicht.

Nur zur Verinnerlichung: durch diese Handlungsweise hast Du gute Chancen, Dein System komplett zu zerschießen, oder Dich auszusperren. Dann heißt‘s eben Daten sichern und neu installieren. Nicht gut.

Was solltest Du aus diesen Zeilen für Dich mitnehmen? Du musst immer wissen, was Du tust. In Windows, Linux und natürlich auch bei Apple.

Bevor Du irgendwo herumdoktorst, leg‘ Dir eine Strategie zurecht.

Ohne Plan oder nur auf *„schau ma amoi…“* drauf los zu klicken…das wird nix. Glaub‘s mir, bringt nur noch mehr Kopfweh.

Du brauchst einen Plan. Egal, was Du wo auch immer herumtust bei den Berechtigungen, es funktioniert alles nach einem Prinzip.

Basisstrategie bei Berechtigungsproblemen

Also, die einzige Strategie, die immer funktioniert bei Berechtigungsproblemen:

Wenn Du wo keinen Zugriff hast,

1) Übernimm den Besitz der Datei, der Festplatte, des Ordners… was auch immer.

2) Vergib die entsprechenden Berechtigungen.

Merke Dir diese 2 Sätze gut. Wenn Du wo „hängst“, rufe Dir die Zeilen ins Gedächtnis. Arbeite sie dann chronologisch ab. Das funktioniert immer… und lass Dich nicht von dem hysterischen Gekreische Deines LAP (Lebens-AbschnittsPartners)… *„wieso gätn des net, sche**ss Windows…in ixpäh hot des funktioniert…“* verrückt machen.

Noch einmal fürs Verständnis:

Nur der Besitzer kann Rechte vergeben. Wenn Du kein Besitzer (einer Datei/Ordner…was auch immer) bist, kannst Du auch an den Berechtigungen nichts ändern.

Für Supporter ist Windows 7 in dieser Hinsicht sicher ein Segen. Bis XP waren die Berechtigungsprobleme im Privatbereich vernachlässigbar, was sich aber seit der Einführung von Vista (und dem einhergehenden neuen Berechtigungskonzept) grundlegend geändert hat.

Um es kurz zu machen: Dein XP-Know-How ist in Windows7 unbrauchbar und deshalb (großteils) für die Rundablage. (Gilt für Privatuser.)

Weitere unbekannte Fehlerquellen können Updates, Tuningtools, Virenscanner, Firewall, Defender… oder eine unglückliche Konstellation dieser Komponenten darstellen.

Besitzrechte in Windows

Wer heutzutage auf einem Windows-System arbeitet, wird standardmäßig ein NTFS-Dateisystem installiert haben.

NTFS ist seit 1993 der Standard für performante Dateiverwaltung in der Windows-Welt. Windows-NT hieß das damals. Parallel gab es noch Windows 95, 98, ME, die alle auf dem „alten“ FAT32-Dateisystem aufbauten.

Ab 1993 (Windows-NT) war NTFS der obligatorische Dateisystemstandard der NT-Linie… Mir fällt auch heute kein plausibler Grund ein, warum man zugunsten von FAT32 auf NTFS verzichten sollte.

Also, wenn wir bei Windows von Rechten sprechen, meinen wir immer die NTFS-Rechte.

Wie funktionieren NTFS-Rechte?

NTFS ist entwickelt worden, weil die bis dahin gängigen Dateisysteme FAT und FAT32 den professionellen Ansprüchen nicht mehr genügten. Beispielsweise war die maximale Kapazität einer Festplatte unter FAT auf 2 GB und unter FAT32 auf 32 GB beschränkt.

Mit NTFS-Berechtigungen ist es nun möglich, auch den lokalen Zugriff zu steuern, bzw. freizugeben oder zu verhindern. NTFS kann aber noch mehr, zum Beispiel können Freigabeberechtigungen, typisch für Netzlaufwerke, nur auf Ordner angewendet werden.

Mit NTFS ist es jedoch möglich, jede einzelne Datei mit NTFS-Rechten zu versehen. Man sagt auch, mit NTFS-Rechten kann man „granularer“ einstellen.

In Firmenumgebungen handhabt man es meist so:

Zuerst vergebe ich Freigabeberechtigungen auf ein Objekt… *(was geb' ich frei im Netzwerk)*

Und mit den NTFS-Berechtigungen regle ich den Zugriff darauf… *(wer darf was)*

Typische Fragen der Rechteverwaltung

Wieso aktiviert sich der Schreibschutz einer Datei immer wieder, obwohl ich ihn deaktiviert habe?

Seit der Einführung von NTFS als Dateisystem sind automatisch alle Dateien schreibgeschützt. Je nachdem, wie die Berechtigungen gesetzt sind, haben die User dann Zugriff auf die Datei.

Wenn Du auf diese Datei nicht zugreifen kannst, liegt das nicht am aktivierten Schreibschutz, sondern an den fehlenden NTFS-Berechtigungen.

Ich habe einen PC mit Dualboot (Windows+Linux) und möchte von Linux aus auf eine NTFS-Partition zugreifen können. Geht das?

Sicher, greif mit dem Finger hin am Bildschirm. Fühlst Du Dich jetzt wohler?…*Spaß beiseite…*

Installiere in Linux das Programm „`ntfs-3g`". Normalerweise ist es aber bei den meisten Linux -Distributionen schon defaultmäßig vorinstalliert. Damit kannst Du lesend und schreibend auf NTFS-Partitionen zugreifen. Für Produktivumgebungen ist dieser Weg nicht empfohlen, für Privatuser reicht er aber allemal.

Ich möchte von Windows auf Linux-Partitionen zugreifen. Geht das?

Vergiss es. Microsoft unterstützt keine anderen Dateisysteme. Es gibt zwar externe Programme, die so was versprechen… aber ein praktikables ist mir bisher nicht untergekommen.

Sieh mal in der Datenträgerverwaltung bei Windows nach, was er da bei Linux-Partitionen ausgibt > unbekannte Partition.

Wie viele Administratoren gibt es überhaupt?

Windows

Seit Vista hast Du einen „Hauptadministrator“ (ist versteckt)

Sonst gibt es beliebig viele Administratoren (die haben erhöhte Userrechte)

Linux

In Linux heißt der Hauptadministrator „root“

root kann so viele andere „roots“ anlegen, wie er will. (macht man zwar nicht, könnte man aber theoretisch…)

Üblicherweise gibt‘s nur einen „root“ auf einer Linux-Box.

Wie erkennt ein Linux-System „root”?

root hat immer die UID *(user id)* 0

Wie aktiviere ich in einer Windows-Home-Version den Administrator?

Da in diesen Versionen der Zugang zu den Benutzerkonten über die Systemsteuerung nicht vorgesehen ist, musst Du den „Administrator“ auf der Eingabeaufforderung freischalten:

Eingabeaufforderung im Admin-Modus starten> `Strg+Shift+Enter` >Kommando>

```
net user Administrator /active
```

Wie starte ich auf einer Windows-Home-Version mit dem Hauptadministrator?

Im abgesicherten Modus…beim Start `F8` drücken…anders geht‘s nicht.

Ich möchte unter Windows eine FAT32-Partition anlegen. Wie groß kann ich die machen?

32 GB. Wenn Du eine größere herstellen willst, nimm ein externes Programm oder eine Linux-CD.

Was ist der Unterschied zwischen Freigabe- und NTFS-Berechtigungen?

Freigabeberechtigungen regeln die Zugriffe über das Netzwerk. NTFS-Berechtigungen regeln die lokalen Zugriffe auf dem PC.

Wenn ich mich lokal am Windows-Computer anmelde, welche Berechtigungen gelten dann?

Die NTFS-Berechtigungen, wenn Du auf einem NTFS-Dateisystem arbeitest. Was heutzutage defaultmäßig eingestellt sein sollte.

Ich habe Windows7 über Vista upgegradet und denselben Usernamen verwendet. Trotzdem habe ich auf verschiedenen Ordnern Berechtigungsprobleme. Was kann ich tun?

Du könntest mal den Müll runtertragen. Trägt zwar nicht zur Lösung Deines Problems bei, aber heh… es gibt wichtigere Sachen als den PC :-)

Der Upgrademechanismus funktioniert in Windows meist. Bei der Rechtezuweisung nach so einer Aktion kann es aber immer wieder zu Problemen kommen. Wenn Du die Wahl hast, bevorzuge immer eine saubere Neuinstallation. Damit schließt Du schon mal potentielle Fehlerquellen im Vorfeld aus.

Falls Du upgegradet hast, eigne Dir Grundkenntnisse der Rechteverwaltung (Besitzübernahme ist schon mal ein guter Anfang…) in Windows an und wende diese an. Weiter unten im Artikel bin ich mal exemplarisch die Besitzübernahme durchgegangen.

Bei mir fehlt unter „Eigenschaften" der Dateien der Reiter „Sicherheit". Wieso?

Du befindest Dich höchstwahrscheinlich auf einer FAT32-Partition. Dort gibt es keine NTFS-Berechtigungen und demnach auch keinen Reiter „Sicherheit", der diese verwaltet.

Oder Du probierst was an deinem Firmen-PC aus, der in einer Domäne „hängt". Dann werden die Berechtigungen nicht lokal, sondern von einer höheren Instanz vergeben. Ergo > Du darfst nichts einstellen auf der Kiste, deshalb gibt's den Reiter *Sicherheit* auch nicht.

Wo kann ich mehr einstellen. In den Freigabe- oder den NTFS-Berechtigungen?

In den NTFS-Berechtigungen.

Ich habe eine externe Festplatte unter XP verwendet. In Windows7 habe ich aber keinen Zugriff darauf. Warum?

In Windows 7 ist der „alte XP-Benutzer" und seine Zugriffsrechte von XP nicht bekannt. Deshalb wird von Windows7 der Zugriff verwehrt. Lösung > Rechte auf der externen Festplatte vergeben.

Obwohl ich Administrator bin, habe ich auf einige Dateien trotzdem keinen Zugriff. Ich dachte, als Administrator darf man alles machen?

Administrator (in Windows) zu sein heißt nicht, alles machen zu dürfen. Es heißt ganz einfach, sich überall Rechte verschaffen zu können.

Schützt mich die UAC vor Schadsoftware?

Nein. Verwende weiterhin Defender, Firewall, Virenscanner und die UAC. Die vier, gepaart mit deinem Hausverstand sind aber ein wirksames Mittel dagegen.

Wie merke ich, dass ich als Hauptadministrator unter Windows arbeite?

Wenn keine UAC aufpoppt, arbeitest Du im Hauptadministrator-Konto. Bei defaultmäßiger Einstellung der UAC.

Worin besteht der Unterschied der UAC in Vista und Windows7?

Vista hat zwei, Windows7 vier Einstell-Level der UAC. Ist also ein bisschen granularer (und benutzerfreundlicher) aufgebaut.

Ist das Hauptadministratorkonto zum Arbeiten gedacht?

Nein, gewisse Dinge funktionieren nicht (Hardwarebeschleunigung…) in diesem Konto. Verwende es nur zu Wartungszwecken.

Ich bin Mitglied in mehreren Gruppen. Welche Rechte gelten jetzt?

Immer die strengeren. Wenn Du in einer Gruppe lesen und schreiben darfst, in einer anderen nur lesen > dann ergeben sich die effektiven Rechte für Dich: nur lesen.

Merke: Verbieten ist immer stärker als erlauben

Die Geschichte der UAC

Seht es mir nach, dass ich ein paar Begriffe aus einem bekannten Italowestern von Sergio Leone zur Veranschaulichung hier angeführt habe. Das lässt das Ganze vielleicht noch ein bisschen plakativer erscheinen.

PS: *„the good, the bad and the ugly"* hieß der Originaltitel des Westerns.

Die Benutzerkontensteuerung hat ihre Wurzeln im Ursprungsdesign von Windows NT. 1993 wurde mit diesem Release ein praktikables Konzept der Authentifizierung realisiert. Beim Einloggen eines Benutzers wird seinem Konto ein Token zugeteilt, der in der gesamten Arbeitssitzung darüber entscheidet, auf welche Daten und Funktionen der Benutzer zugreifen kann. Der Token enthält die SID *(Security ID)* des Benutzers sowie dessen zugehörige Gruppen. Bei jedem Zugriff auf eine Funktion prüft das System den Token, ob die angeforderte Aktivität erlaubt ist. Diese Technik ist sehr effizient und praktikabel, da nur die SIDs in der Zugriffsliste *(Access Control List, ACL)* mit dem Token verglichen werden.

Änderungen an Gruppenmitgliedschaften werden so aber nur wirksam, wenn ein Benutzer sich ab- und neu anmeldet. Auch klar.

Jede Anwendung, die von einem Benutzerkonto gestartet wird, bekommt das Token des Benutzers übertragen. Dadurch hat eine Anwendung nicht mehr Rechte als der Benutzer, der es aufruft.

Der Zugriff auf Systemfunktionen wird von Windows über die Mitgliedschaft in Gruppen gesteuert (ditto in Linux): Nur wer Mitglied der lokalen Gruppe *„Administratoren"* eines Systems ist, hat tatsächlich Administratorrechte. Bei Windows7 bist Du automatisch als *„eingeschränkter Administrator"* angemeldet nach einer Neuinstallation. Bei XP warst Du immer automatisch ein vollwertiger Administrator.

Die Praxis der letzten Jahre verdeutlichte uns die Bequemlichkeit dieser Tatsache. Eine Menge Administratoren, Benutzer und Softwareentwickler arbeiteten hauptsächlich mit erhöhten Rechten. Als Administrator eben.

Das heißt, das Windows98 Syndrom, das ein Einzelplatzsystem mit entsprechenden Rechten überall darstellte, wurde man so leicht nicht los. Wenn man die User über Jahre einmal daran gewöhnt hat, überall am System defaultmäßig uneingeschränkten Zugang zu haben (bis einschließlich XP), war ein Zurückrudern sehr aufwendig.

Microsoft entschied sich daher bei der Entwicklung von Windows Vista für einen ganz neuen Weg: Die Benutzerkontensteuerung erzwingt, dass alle Benutzer ohne Administratorrechte arbeiten – selbst dann, wenn sie Mitglied der Gruppe *„Administratoren"* sind. (das führt bei vielen Usern oft zur Verwirrung…)

UAC

Sollte es tatsächlich notwendig sein, eine Aktivität mit Admin-Rechten auszuführen, so bietet UAC dies an.

Sobald nun das System feststellt, dass eine Anwendung nur dann gestartet oder fortgesetzt werden kann, wenn Admin-Rechte vorhanden sind, hält es die Ausführung an und fragt den Benutzer, ob er sich diese Rechte verschaffen will. Da Vista auf demselben Code wie Server 2008 aufbaut, ist ein grundsolides Wissen der UAC auch für Netzwerkadministratoren unabdingbar. **Bild 1**

Die UAC ist ein Baustein in einem umfassenden Sicherheitskonzept, das aber immer um weitere Komponenten wie Virenscanner, eine Firewall und dem entsprechenden Benutzerverhalten ergänzt werden muss.

In diesem Moment kommt die (als nervig empfundene) UAC-Nachfrage: Windows blendet ein Dialogfenster ein. Der Benutzer kann in dieser Situation nichts anderes tun und muss die Frage beantworten.

Je nachdem, wie weit der für den Rechner verantwortliche Admin das Sicherheitskonzept bereits verinnerlicht hat, kann eine von zwei Situationen vorliegen:

Das angemeldete Benutzerkonto ist ein „normales" Benutzerkonto, das gar nicht Mitglied der Administratoren-Gruppe oder einer anderen Gruppe mit erhöhten Rechten ist. Oder aber das Konto hat eigentlich die nötigen Rechte, aber durch das von der UAC beschränkte Token kann es diese Rechte nicht einsetzen. Daraus resultieren unterschiedliche Abfragedialoge:

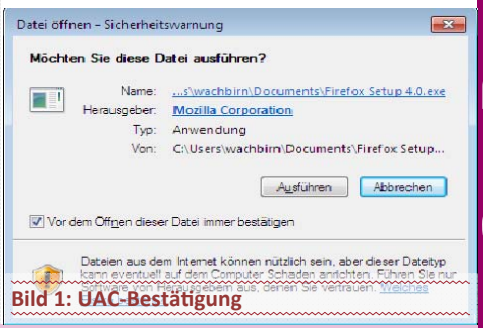


Bild 1: UAC-Bestätigung

The good (der Gute): Benutzer ist kein Admin

Der Benutzer hat ein „normales“ Benutzerkonto. Da einem normalen Benutzerkonto nicht mal so eben Admin-Rechte zugeschanzt werden können, bietet die aufpoppende UAC zwei Eingabefelder an: Name und Passwort eines Kontos, das über die nötigen Rechte verfügt… üblicherweise das Admin-Konto. Sind die Eingaben korrekt, startet das System mit dem „neuen“ Token des eingegebenen Namens. **Bild 2**

Die Philosophie dahinter entspricht im Wesentlichen der *„Run as“*-Funktion *(„Ausführen als“)*, die mit Windows 2000 Einzug hielt. Hauptverantwortlich dafür ist übrigens der Windows-Dienst *„Sekundäre Anmeldung“*.

So weit so gut. Was hierbei gerne vergessen wird, ist die Tatsache, dass kein normaler Benutzer Zugang zum Kennwort des Admins haben sollte. Die ureigenste Intention dieser Aktion ermöglicht einem anwesenden Administrator, einem Benutzer zu helfen, indem er sich einmalig authentifiziert und eine Aktion mit erhöhten Rechten ausführt.

Diese Technik ist im Fachjargon auch unter dem Namen *„Over-the-shoulder (OTS)“* bekannt, weil der Administrator dem Benutzer gewissermaßen über die Schulter hinweg hilft.

Der interessante Lerneffekt für den Administrator ist natürlich auch nicht von der Hand zu weisen. Heutzutage hat jeder gewissenhafte Admin zumindest zwei Konten eingerichtet. Ein normales Benutzerkonto für alltägliche Arbeiten und eines für administrative Arbeiten. (ditto in Linux). Der gangbare Weg sollte immer der sein, sich mit einem normalen Benutzerkonto am System anzumelden, und je nach Bedarf mittels UAC für eine administrative Aktion erhöhte Rechte zu erlangen.

Wenn aber keine erhöhten Rechte verlangt werden, ist die UAC ein komfortabler Ersatz für „runas“.

Wie wir später sehen werden, gibt es diese Möglichkeit nur bei OTC und nicht bei AAM *(Administrator Approval Mode…Administrator-mode bestätigen)*.

The bad (der Böse): Benutzer ist Admin

Jetzt wird’s interessant. Der Benutzer sollte eigentlich über keine lokalen Adminrechte verfügen. Da er aber ab Vista diese automatisch zugeteilt bekommen hat, wurde mit der Entwicklung der UAC diesem Umstand Rechnung getragen. Der Token wird um die administrativen Rechte bereinigt, sodass nur die „benutzerspezifischen“ Attribute vorhanden sind. Kommen wir jetzt zu einer Situation, wo wir erhöhte Rechte brauchen, poppt wieder die UAC auf. Aber diesmal mit einem anderen Dialog. Der fordert „nur“ zur Bestätigung der Aktion auf. **Bild 3**

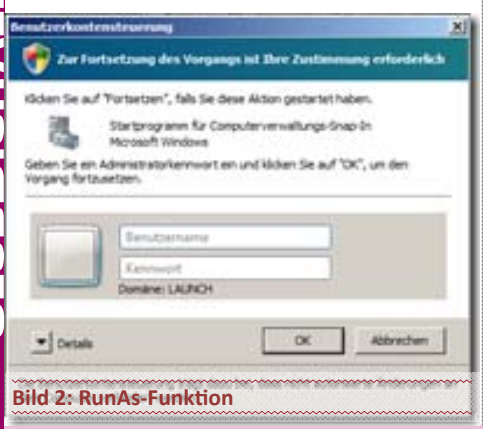


Bild 2: RunAs-Funktion

Warum ist das so?

Wir haben ja ausreichend Rechte, da wir ja in der Gruppe der Administratoren sind? Der einzige Grund für diese Aktion ist die Bestätigung des Administrators, dass er diese Aktion angefordert hat > *Admin Approval Mode* eben.

Jetzt gibt es auch hier wieder zwei Szenarien. Ist die aufgerufene Applikation gut an die UAC angepasst, gibt diese sie anstandslos frei. Wenn die Anwendung aber nicht mit der UAC umgehen kann, muss mit einem neuen Token (der natürlich Admin-Rechte hat), die Aktion neu gestartet werden.

Wie mache ich das? Mit rechter Maustaste auf das gewünschte Programm gehen und im Kontextmenü *„Als Administrator ausführen“* auswählen.

The ugly (der Hässliche): Der „Administrator"

Die Benutzerkontensteuerung muss mit einer Ausnahme leben. Dem *„Administrator“*. Das vordefinierte Konto *„Administrator“* ist das einzige Konto auf einem Windows-System (ab Vista), dass niemals durch UAC eingeschränkt wird. Darum ist es wahrscheinlich auch standardmäßig versteckt.

Bei den Home-Versionen von Windows7 kannst Du nur im abgesicherten Modus mit diesem Konto starten.

Was Du Dir merken musst:

Bis Windows XP warst Du automatisch als „Hauptadministrator *(the ugly)* angemeldet am System.

Ab Vista bist Du als Benutzer mit eingeschränkten Adminrechten *(the bad)* am System angemeldet. Damit das funktioniert, wurde die UAC ins Leben gerufen.

Der Hauptadministrator *(the ugly)* umgeht die UAC. Dieses Konto ist aus diesem Grund standardmäßig deaktiviert. Damit Du nicht unbeabsichtigt etwas am System „zerstören" kannst.

Jeder neu angelegte Benutzer ist automatisch der Gruppe *„Administratoren“* zugeteilt, die wiederum in den Wirkungsbereich der UAC fällt.

Aufpassen > Administratoren = eine Gruppe

Administrator = „der“ Chef am System

Das gilt uneingeschränkt für Desktopsysteme ab Windows-Vista, nicht aber für Windows-Serverversionen. Dort ist‘s ein bisschen anders…

Die UAC in der Praxis

Für viele Umsteiger von Windows XP ist die Benutzerkontensteuerung ein

suena a chino (spanisches Dorf)

Die Foren sind voll von Beschwerden, dass man trotz angemeldetem Administrator keinen Vollzugriff auf alle Dateien im System hat oder dass in der Eingabeaufforderung manche Befehle nicht funktionieren. Bei der Installation von Anwendungen poppt einmal eine Sicherheitsabfrage auf, ein anderes Mal erscheint überhaupt keine Abfrage und die Installation bricht ab, usw…

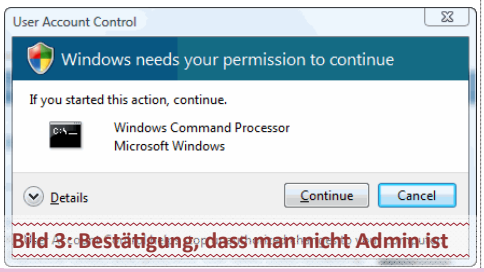


Bild 3: Bestätigung, dass man nicht Admin ist

Was ist das Problem?

Grundsätzlich ist es wichtig, zu verstehen, wie Windows ein Benutzerkonto verwaltet.

Meldet sich ein Standardbenutzer am System an, erhält er ein Sicherheitstoken für Standardnutzer zugewiesen. Ein Standardbenutzer kann sich mit Hilfe der Benutzerkontensteuerung ein Sicherheitstoken des Administratorkontos für administrative Aufgaben zuteilen lassen.

Meldet sich ein Administrator am System an, erhält er zwei Tokens. Ein Token steht für normale Benutzerberechtigungen und wird beim Arbeiten unter dem Administratorenkonto standardmäßig verwendet. Das zweite Token besitzt Administratorenrechte, muss aber von der Benutzerkontensteuerung freigegeben werden. Es wird immer mit dem niedrigstwertigen Token gearbeitet.

Die Benutzerkontensteuerung (UAC) besitzt also die Fähigkeit, die Berechtigungen des Administrators (oder eines Standardbenutzerkontos) auf die erhöhte Stufe anzuheben.

Bei den meisten Windows-Systemfunktionen erfolgt der Aufruf der Benutzerkontensteuerung automatisch und nach Bestätigung kann der Vorgang durchgeführt werden.

Liefert aber der Entwickler einer Anwendung, die Administratorenrechte benötigt, die entsprechenden Routinen mit, kann Windows 7 diese auswerten (zum Beispiel `setup.exe`…) und automatisch das Dialogfeld der Benutzerkontensteuerung aufrufen. Das wäre der Idealfall. Funktioniert aber nicht immer.

Bei manchen Softwareprogrammen unterbleibt beim Aufruf die Anfrage der Benutzerkontensteuerung. Dadurch arbeiten wir mit unserem Benutzerkonto-Token weiter, anstatt einem benötigten Admin-Token.

Folglich haben wir dann Schwierigkeiten, Operationen auszuführen, die Adminrechte erfordern.

Dasselbe Szenario haben wir beim Aufruf der Eingabeaufforderung. Auch dort kommt keine Anfrage der Benutzerkontensteuerung. (Was ich dort aber eher vermisse, ist die klare visuelle Erkennung, ob diese Shell jetzt erhöhte Rechte hat oder nicht.)

Eine Möglichkeit wäre, die Benutzerkontensteuerung abzuschalten und ständig unter einem Administratorenkonto zu arbeiten. Das bringt aber erfahrungsgemäß andere Probleme mit sich. Ich rate deshalb auch strikt von dieser Vorgehensweise ab.

Prinzipiell gibt es zwei Möglichkeiten, wie man zum Beispiel Systemdateien editieren kann oder Zweige der Registry. Aus einem Benutzerkonto heraus natürlich.

Leg‘ eine Verknüpfung an und setz deren Eigenschaften auf *Als Administrator ausführen*.

Klicke die Programmdatei oder den Startmenüeintrag mit der rechten Maustaste an und wähle den Kontextmenübefehl *Als Administrator ausführen*. Einfacher und komfortabler geht‘s wirklich nicht mehr.

In beiden Fällen erscheint das Dialogfeld der Benutzerkontensteuerung und nach einer Administratorenfreigabe kann Windows die betreffende Anwendung mit dem Administrator-Token ausführen.

Bei älteren Windows-Versionen gibt‘s noch die Chance, den Administratormodus auf der Registerkarte *„Kompatibilität"* zu erzwingen. Die Anwendung läuft dann im ganzen System immer als *„Administrator“*.

Windows Explorer

Ein unrühmlicher Punkt der UAC ist derweil noch der Windows Explorer. Warum? Schon mal versucht, von einem Standardbenutzerkonto aus den Kollegen im Administratorenmodus zu starten? Viel Glück.

Das wär's doch! Du hättest überall grafisch auf sämtliche Files Zugriff, ohne dass jetzt jede Datei einzeln als Admin aufwerten zu müssen (falls das überhaupt funktioniert…) oder in der Eingabeaufforderung zu tricksen. Kannst editieren und kopieren, was das Zeug hält. Wenn man fertig ist, verlässt man den Dateiexplorer im Administratormodus.

Die Explorergeschichte im Administratormodus, initiiert vom Benutzerkonto aus funktioniert hervorragend. Leider nur in Linux.

Keine Ahnung jetzt, ob das am Systemaufbau der Userverwaltung von Microsoft oder der Windows-Shell liegt. Ärgerlich ist es allemal.

Leider basiert die Windows-Shell auf dem Windows-Explorer. Das verwirrt auch anfangs viele, die einen Windows-Server nur mit der Windows-Shell installieren möchten. Geht nicht. Du brauchst unbedingt den grafischen Unterbau, damit die Windows-Shell lauffähig ist.

Bei Linux ist es genau umgekehrt. Da hast Du einmal die Shell. Die hast Du immer, sobald Du irgendein Linux startest. Keine Shell, kein Linux. Von dem ganzen „Linux-Shell" Zeugs weg wird dann die Grafik (wenn erwünscht) gestartet.

Sprich, Linux ist weitaus modularer aufgebaut. Deshalb gibt's ja auch so viele Konstellationen und Möglichkeiten, wie man ein System betreiben will. Der User entscheidet.

So, aber jetzt wieder zurück zu unserem Windows-Explorer…

Na super, und jetzt? Was mach ich jetzt in Windows?

Du kannst zwar neue Instanzen der `explorer.exe` als einzelne Prozesse im Administratormodus ausführen. Es erscheint auch die Benutzerkontensteuerung, die Du auch abnicken kannst. Und dann schau mal in den Taskmanager hinein. Diese Kopie der aufgerufenen Shell läuft trotzdem weiterhin mit normalen Benutzerrechten!

Mensch, ich hab geschaut wie ein Schwein in ein Schweizer Uhrwerk…wieso geht des net?

Im Administratorkonto den Explorer aufrufen? Nicht sehr praktisch, jedes Mal das Konto zu wechseln.

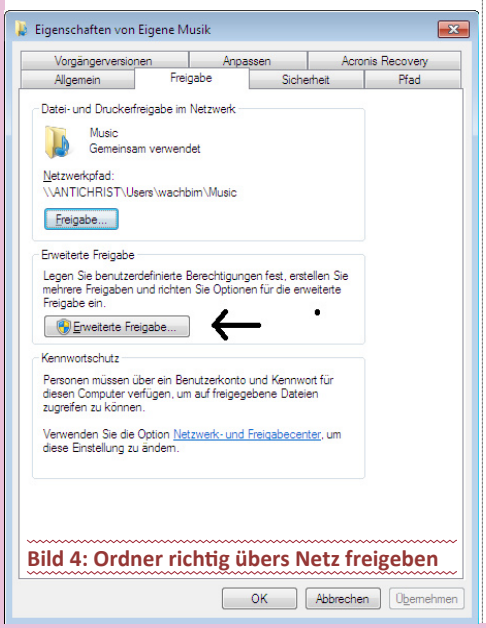


Bild 4: Ordner richtig übers Netz freigeben

Die Lösung

Verwende einfach einen anderen Dateimanager, der nicht die Bereitstellung der Windows-Shell (oder einer API-Schnittstelle) voraussetzt.

Ich verwende dafür den „A43"-Dateimanager. Es gibt aber genug andere auch noch. (Explorer++…) Einfach googeln.

Den A43 einfach per Rechtsklick *„Als Administrator ausführen"* starten, und schon sind wir wieder im Spiel und haben Vollzugriff. Auch auf Dateien, auf die ein Admin normalerweise keinen Zugriff hat.

Ein weiteres bekanntes Berechtigungsproblem unter Windows ist der verweigerte Zugriff auf die Ordner *„Dokumente und Einstellungen"*. Auch als Administrator.

Lösung: *„Dokumente und Einstellungen" (documents and settings)* sind keine Ordner, sondern NTFS-Links. Wenn Du genau schaust, haben die links unten einen Verknüpfungspfeil. Diese Verknüpfungen sind aus Abwärtskompatibilitätsgründen *(was für eine Wortschöpfung, Hartl)* vorhanden, da bei Windows 7 das Profil eines Benutzers im Pfad *„Users"* (und *ProgramData*) liegt, und nicht wie früher unter *„Dokumente und Einstellungen"*.

Sobald ein „älteres" Programm unter *„Dokumente und Einstellungen"* (was es ja in Windows7 nicht mehr gibt) hineinzuschreiben versucht, wird es automatisch nach `„C:\Users"` umgeleitet.

Schalte einfach die *„Anzeige von versteckten Systemdateien"* aus, dann siehst Du die NTFS-Links auch nicht mehr. Es ist speziell in Windows 7 eine gute Idee, versteckte Dateien nicht anzuzeigen. Was man nicht sieht, macht kein Kopfweh.

Freigabeberechtigungen…

Hier hat sich auch einiges geändert gegenüber XP. Interessant für Leute, die ein Heimnetzwerk haben.

Konkret geht es um die Ordnerfreigabe im Netzwerk.

Grundsätzlich: Ordnerfreigaben unterscheiden sich von NTFS-Berechtigungen. Klar wurden diese zwei Sachen auch in früheren Windows-Versionen getrennt. In Windows 7 kann man aber beide Berechtigungen (meist unwissentlich) gleichzeitig ändern. Oft kommt man dann nicht mehr an die Daten ran.

Gehen wir's durch:

Klicke mit der rechten Maustaste auf den Ordner, den Du freigeben willst. Dann auf *„Eigenschaften"* natürlich.

Im Eigenschaftsfenster siehst Du zwei Register. *Freigabe* und *Sicherheit*. *„Freigabe"* regelt Zugriffe übers Netzwerk, *„Sicherheit"* regelt NTFS-Rechte lokal am System. War auch schon früher so. **Bild 4**

Da wir einen Ordner über das Netzwerk freigeben wollen, gehen wir auf den Reiter *„Freigabe"*.

Du siehst dort drin einmal *„Freigabe"* und dann unten noch einmal *„erweiterte Freigabe"*.

Wenn Du jetzt aber in Deiner logischen Annahme auf *„Freigabe"* drückst, um den Ordner im Netzwerk freizugeben, wirst Du keine Freude haben. Dieses Vorgehen ändert nämlich beide (die NTFS- und die Freigabeberechtigungen für das Netzwerk gleichzeitig). Berechtigungsprobleme sind da vorprogrammiert. Das muss man wissen und zur Kenntnis nehmen.

Falls Du „nur" einen Ordner freigeben willst, musst Du immer den unteren Button *„Erweiterte Freigabe"* nehmen. Das ist der Knackpunkt.

Alles Weitere kann man getrost in XP-Manier fertigstellen. Wann und wofür man den anderen Button *„Freigabe"* verwenden sollte, kann ich dir ehrlicherweise jetzt auch nicht sagen.

Feedback in diese Richtung kann auf jeden Fall nicht schaden. Selbst meine Mutter die keine Ahnung von der EDV hat, konnte mir da nicht weiterhelfen. Hm...

Ein Tipp noch: Standardmäßig ist die Gruppe *„Jeder"* für den Zugriff berechtigt. Sinnvoller wäre es aber, einzelnen Gruppen den Zugang zu ermöglichen. Tu das. Wenn Du dann den Gruppen Rechte vergibst, beachte die Faustregel: Immer mit Bedacht verweigern. Lieber nicht so viele Rechte einer Gruppe zuschanzen (zulassen), als einer Gruppe viele Rechte entziehen (verweigern).

Der Grund: Negative Freigaberechte setzen die positiven Freigaberechte immer außer Kraft. Sprich: Verweigern ist immer stärker als Zulassen. So vermeidet man schon im Vorfeld Berechtigungsprobleme.

Windows weigert sich, Setup-Programme auszuführen… **Bild 5**

Das kann man leicht fixen. Wenn man's weiß. Aber es steht ja eh recht deutlich auf der Dialogbox. *>…durch die Internetsicherheitseinstellungen wurde verhindert…*

Da haben wir normalerweise schon den Übeltäter. Öffne *IE > Extras > Internetoptionen > Sicherheit > Internetzone > Stufe anpassen*

Im Dialogfeld Sicherheitseinstellungen-Internetzone geh zum Zweig *Verschiedenes > Anwendungen und unsichere Dateien starten*…dort *bestätigen* markieren, das war's.

Besitzer

Jeder Ordner und jede Datei muss einen Besitzer haben. Immer. Der Besitzer ist derjenige, der das Objekt erstellt hat. Üblicherweise den Ordner oder die Datei.

Nicht jedes Objekt wird von einem „echten" Benutzer erstellt. Beispielsweise werden bei der Installation des Betriebssystems oder der Installation von Programmen ja auch automatisch Ordner und Dateien erstellt. Die sind natürlich auch einem Besitzer zugeordnet.

Der Besitzer wird im Register *Besitzer*, in den *Erweiterten Sicherheitseinstellungen* unter *Aktueller Besitzer* angezeigt.

Der Besitz ist eine wichtige Eigenschaft (wenn nicht die wichtigste) bei den NTFS-Rechten. Denn der Besitzer stellt die oberste Instanz bei der Vergabe von Berechtigungen dar. Sprich, der Besitzer bestimmt, wer was machen darf.

Das ist deshalb wichtig, weil es durchaus möglich ist, festzulegen, dass zum Beispiel niemand eine Berechtigung auf das Objekt bekommen soll.

Was geschieht aber, wenn niemand auf das Objekt zugreifen kann? Weil niemand auf das Objekt zugreifen kann, können wiederum keine neuen Berechtigungen vergeben werden.

Sogar der Administrator kann nicht darauf zugreifen. Das führt manchmal zur Verwirrung. Denn viele Administratoren sind verwundert

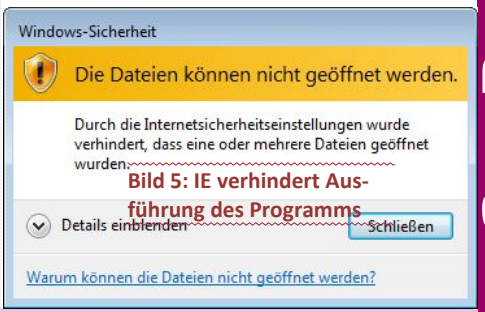


Bild 5: IE verhindert Ausführung des Programms

darüber, dass sie auf ein Objekt nicht zugreifen dürfen, obwohl sie ja *„Administratoren"* sind.

Egal, wenn Du keine Rechte hast, ja…dann kannst Du auch nicht zugreifen. Punkt.

Maßgeblich für den Zugriff sind die Berechtigungen. Wer nicht berechtigt ist, kann auch nicht zugreifen. Nicht einmal der Ersteller! Das nennt man *„sich aussperren"*. Merke Dir den Satz gut. Wenn Du Dich trotzdem ausgesperrt hast…lies weiter.

Der Ersteller kann jedoch eines tun. Der Ersteller kann als letzte Instanz neue Berechtigungen vergeben. Aus dem Grunde hat grundsätzlich jedes Objekt einen Besitzer.

Das ist extrem wichtig für das Verständnis, dass jedes Trum einen Besitzer hat. Auch wenn es kein „User" ist…irgendwem *„gehört"* die Datei. Und dort muss man dann einhaken.

Der Besitzer kann die Berechtigungen einsehen und neu vergeben. Erst danach können weitere Benutzer die Berechtigungen einsehen und je nachdem was eingestellt wurde, selbst weitere Berechtigungen vergeben.

Conclusio > Nur der Besitzer kann Rechte vergeben auf das jeweilige Zeugs.

Was passiert aber, wenn es den Besitzer nicht mehr gibt? Auch für solche Fälle gibt es eine Lösung. Ein Administrator hat die Möglichkeit, den Besitzer so zu ändern, dass entweder der Administrator oder die Gruppe der Administratoren zum Besitzer ernannt wird.

Sprich, er übernimmt das Zeug (der Administrator).

Auch in der täglichen Arbeit kann es notwendig sein, dass der Administrator kurzfristig einen Besitz übernehmen muss. Beispielsweise haben die Administratoren keine Berechtigung, auf alle Benutzerprofile zuzugreifen. Manchmal kommt es jedoch vor, dass der Administrator darauf zugreifen muss.

In solchen Fällen kann der Administrator vorübergehend den Besitzer ändern, sich berechtigen und das Profil supporten.

Hier mal exemplarisch, wie ich den Besitz einer externen Festplatte übernehmen kann. Die externe Festplatte, die unter XP noch tadellos zugänglich war, verweigert unter Windows 7 den Zugriff. Das hängt damit zusammen, dass der XP-User (der ja bisher der Plattenbesitzer war) dem Windows 7-System nicht bekannt ist.

Was wollen wir? Den Besitz einer externen USB-Platte übernehmen

Du machst jetzt am besten alles im Konto *„Administrator"*. Wie Du das aktivierst, steht weiter oben im Artikel.

Arbeitsplatz > rechte Maustaste auf den Datenträger > im Tab *„Sicherheit"* auf *„Bearbeiten"*

Zwei Szenarien sind nun möglich:

Szenario 1

Du siehst hier exemplarisch, dass wir keine Zugriffsrechte besitzen.

Um dies zu ändern, werden wir aufgefordert, auf den Button *„Fortsetzen"* zu klicken. **(siehe Bild 6)**. Das machen wir natürlich…

Nun sehen wir die erweiterten Sicherheitseinstellungen für das Laufwerk. **Siehe Bild 7.** Im Endeffekt siehst Du schon, wie es „fertig" aussehen soll. Besitzer ist die Gruppe *„Administratoren"*. Wie ich das gemacht habe, zeige ich Dir gleich.

Um nun einen Besitzer einzutragen (wir nehmen die Gruppe *„Administratoren"*), klicken wir in der Rubrik *„Besitzer ändern nach"* am besten erst einmal

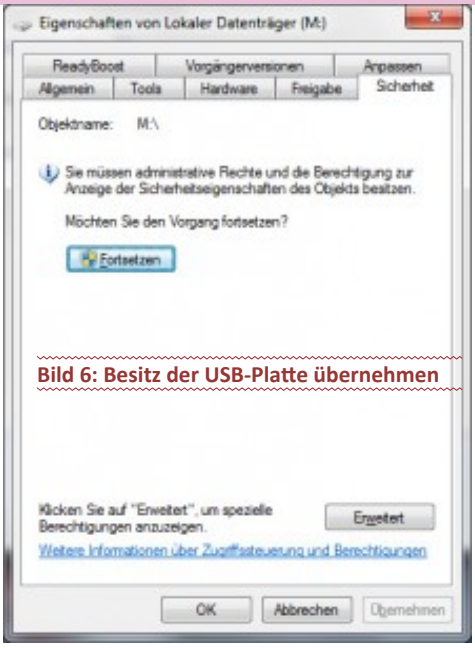
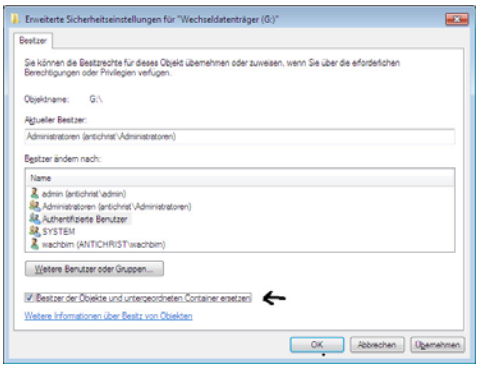


Bild 6: Besitz der USB-Platte übernehmen

Bild 7: Erweiterte Sicherheitseinstellungen



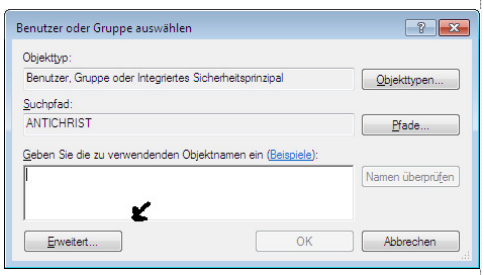


Bild 8: Benutzer und Gruppen auswählen

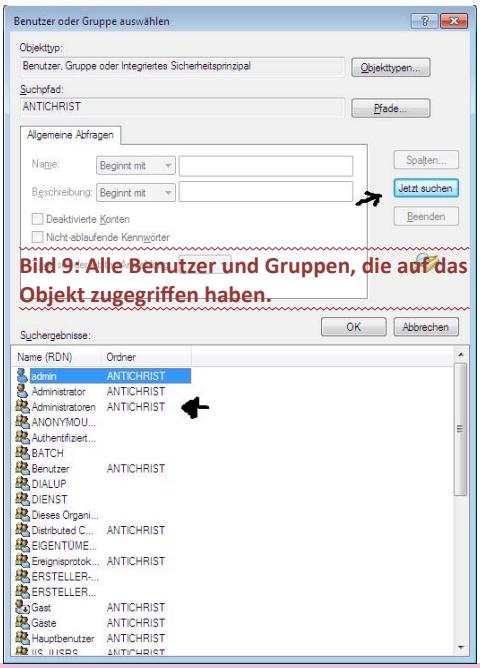


Bild 9: Alle Benutzer und Gruppen, die auf das Objekt zugegriffen haben.

die *„Administratoren"* an. Die würd ich generell immer drin lassen. Wenn Du die weglässt, kann man nur mehr mit vielen Verrenkungen die Rechte wieder gerade biegen. Also schau, dass die Gruppe der Administratoren bei deinen Dateien zumindest immer dabei ist.

Ist diese Gruppe dort nicht zu finden, müssen wir diese unter *„Weitere Benutzer oder Gruppen"* hinzufügen. Logisch… Dann siehst Du einmal das da. **Bild 8**

In dem Auswahlfeld klicken wir hierzu auf den Button *„Erweitert"*. Es erscheint eine neue Box… siehe **Bild 9**. Dort gehst Du rechts auf *„Jetzt suchen"*.

Dann bekommen wir in den Suchergebnissen alle vorhandenen Benutzer und Gruppen angezeigt, die auf dem Rechner vorhanden sind. Siehe **Bild 9** unten. *Meine Freihandpfeile haben's in sich-muss ich schon sagen.*

Danach fügst Du diese 4 Objekte hinzu:

- SYSTEM
- Authentifizierter Benutzer
- Administratoren
- Benutzer

Ich persönlich gebe immer diese 4 Objekte an. Notwendig sind sie wahrscheinlich nicht, aber mit dieser Konstellation bin ich auf der sicheren Seite, falls Windows7 herumzickt…

Merke: Gruppen haben immer zwei Köpfe, Benutzer immer einen.

Schlussendlich, markiere die *„Administratoren"* blau.

Nun klicken wir zweimal auf *„OK"* und sind zurück in der Übersicht. Hier ist noch wichtig, dass wir das Feld *„Besitzer der Objekte und untergeordneten Container ersetzen"* auswählen. **(Bild 7 unten beim ästhetischen Pfeil…)**

Denn so übernehmen wir gleich für den gesamten Inhalt den Besitz. Ohne diesen Haken können wir sonst später zwar auf die Festplatte zugreifen, aber keine enthaltenen Dateien ändern oder löschen. Das wäre nicht so gut…

Szenario 2

Es kann sein, dass Du unter *„Eigenschaften"* und dem Reiter *„Sicherheit"* der Platte diese Maske zugewiesen bekommst. Ob das mit den verschiedenen Rechten zusammenhängt, kann ich Dir ad hoc nicht wirklich genau sagen. Das Ganze ist aber kein wirkliches Problem. Gehe einfach unten auf den Button *„erweitert"* **(siehe Bild 10)**.

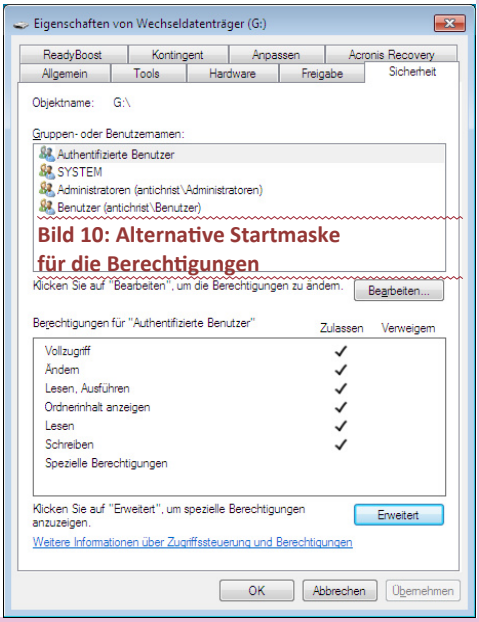


Bild 10: Alternative Startmaske für die Berechtigungen

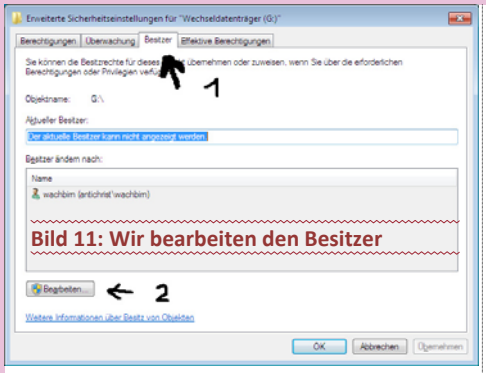


Bild 11: Wir bearbeiten den Besitzer

Bild 12: Sicherheitshinweis zur Rechteübernahme

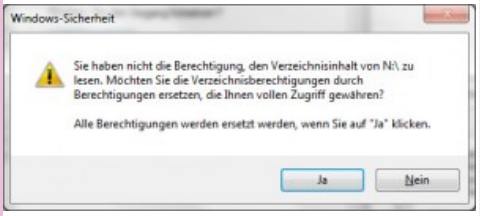


Bild 13:Alles schließen und öffnen damit das System es übernimmt

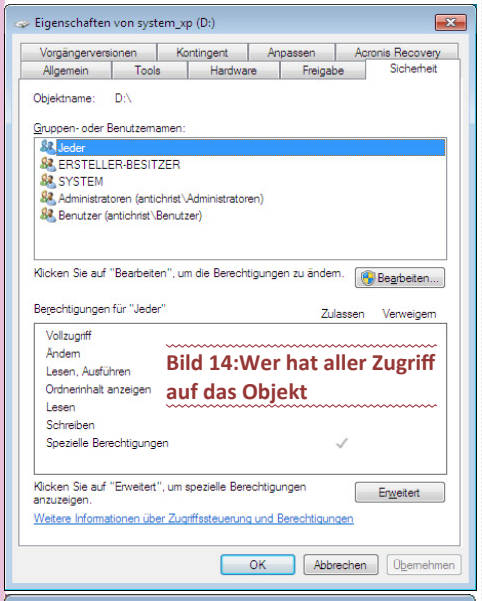


Bild 14:Wer hat aller Zugriff auf das Objekt

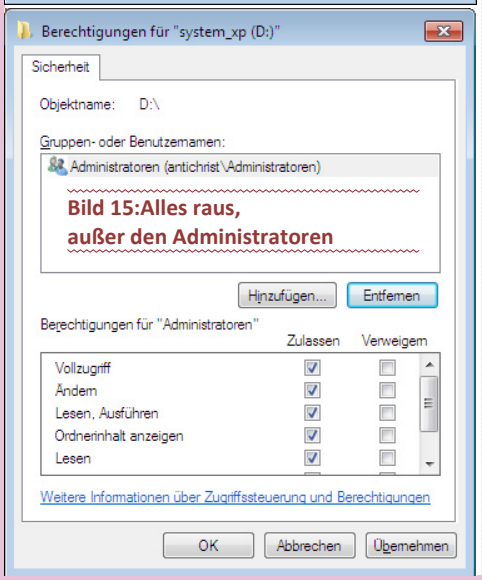


Bild 15:Alles raus, außer den Administratoren

Eine neue Maske kommt… *nona*

Wähle oben den Reiter *„Besitzer“* und dann unten *„bearbeiten“*. **Bild 11**.

bild 11

Jetzt unten auf *„weitere Benutzer und Gruppen“*…und wir sind wieder bei einem alten Bekannten > **Bild 8**…dort geht’s ganz normal weiter wie in Szenario 1.

Nach dem Übernehmen der Maske **(Bild 7)** und zweimaligem Bestätigen geht’s jetzt in beiden Varianten gleich weiter…hoffentlich :-)

Ein neuer Hinweis kommt > **Bild 12.** Drücke auf *ja*…(wir wollen ja die Berechtigungen ersetzen…) jetzt kann es eine Weile dauern, je nachdem wie viele Dateien auf der Platte sind. Nachdem das abgeschlossen ist, kommt noch ein erneuter Hinweis > **Bild 13**

Drücke auf *„ok“* und schließe alle Fenster. So, eigentlich müsste das passen.

Haben wir alle Fenster geschlossen, sollte auch der Zugriff auf die Festplatte wieder möglich sein! Nun können wir prüfen, ob die Inhalte alle richtig angezeigt werden, was nun auch wieder problemlos möglich sein sollte.

Diesen Vorgang kann man übrigens nicht nur bei externen Festplatten nutzen, sondern auch bei intern eingebauten Festplatten, die man zum Beispiel aus einem XP-Rechner in den Windows 7-Rechner umbaut.

Merke: Sicherheit und Benutzerfreundlichkeit schließen sich immer aus. Man muss immer einen Kompromiss finden. In Windows ist die UAC ein Teil davon…

Eine andere Konstellation: Du hast Daten auf einer Partition oder Festplatte liegen, und möchtest, dass nur Du und der Administrator darauf zugreifen können.

Leider unterscheiden sich die Dialoge ein bisschen zum vorigen Szenario, sodass ich die Fotostrecke erneut durchgehen muss…

Besitzübernahme ist wieder mal das Zauberwort.

Merke: zuerst immer den Besitz übernehmen, dann die Rechte zuweisen.

Wenn Du mal keinen Plan hast, erinnere Dich an diesen Satz. Er ist immer die Lösung bei Berechtigungsproblemen.

Gehe wieder ins Administratorkonto

Unter XP hast Du noch die *„einfache Dateifreigabe“* deaktivieren müssen. Brauchst bei Windows7 nicht mehr…

Wir fangen mal groß an und sperren einmal eine Partition.

Merke *(scho wida wos…)*

Bei der Rechtevergabe immer von oben nach unten arbeiten…

Exemplarisch sperre ich jetzt meine XP-Partition Buchstabe D. Gehe dort wieder auf *„Eigenschaften“* und den Reiter *„Sicherheit“.*

Im oberen Feld siehst Du schon mal wer aller Zugriff auf die Partition hat. Analog dazu im unteren Feld die entsprechenden Rechte. **Siehe Bild 14.**

Auch hier gilt wieder: Es ist sinnvoll bei der Bearbeitung der Rechte, die Gruppe der *„Administratoren“* nicht zu entfernen.

Wir wollen allen Benutzern, außer dem eigenen Benutzerkonto und den Administratoren den Zugriff verwehren.

Klicke auf *„bearbeiten“* und lösche alles raus, außer den Administratoren> **Bild 15**

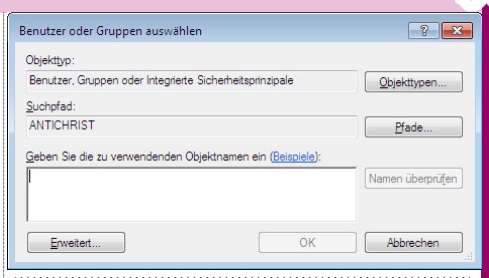


Bild 16:Wieder auswählen, wer zugreifen darf

Je nachdem, wie viele Dateien sich auf der Partition befinden, kann das Übernehmen der neuen Einstellungen einen Moment dauern. Ist dieser Vorgang abgeschlossen, können wir uns testweise mit unserem Benutzerkonto anmelden und versuchen, auf die Partition zuzugreifen, es wird uns nicht gelingen.

Unser Benutzerkonto müssen wir noch hinzufügen. Dazu klicken wir auf den Button *„Hinzufügen“.*

Jetzt kommt wieder ein alter Bekannter > **Bild 16**

Hier können wir entweder den Benutzernamen in das Feld eintragen oder aber auch über den Button *„Erweitert“* sichergehen und den Namen aus der Benutzerliste heraussuchen. Wie im vorigen Szenario beschrieben…

Im nächsten Fenster klicken wir auf *„Jetzt suchen“* und danach werden im unteren Fenster alle verfügbaren Benutzer und Gruppen angezeigt. Wähle auch hier deinen Usernamen aus.

Übernimm das alles wie gewohnt. Danach siehst Du sowas in der Art> **Bild 17**

antichrist = Rechnername

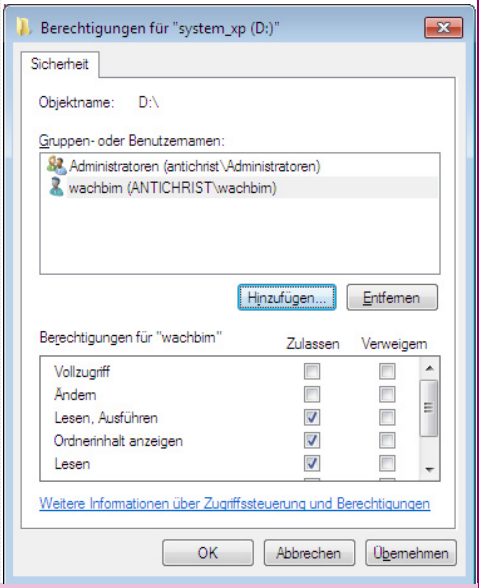
wachbirn = Benutzername

Schauen wir uns nun den Benutzer (`wachbirn`) an fällt uns auf, dass er nicht so viele Rechte hat wie der Administrator. Dies ist auch gut so, denn das unterscheidet den Benutzer vom Administrator. Dennoch sollte man aber hier noch den Haken setzen bei *„Ändern“,* da es sonst mit unserem Benutzerkonto nicht möglich ist, irgendwelche Dateien auf dieser Festplatte zu ändern.

Haben wir dies getan, speichern wir die Änderungen ab und können uns nun nochmals als Benutzer einloggen und wir werden sehen, dass wir nun auch als Benutzer Zugriff auf die Festplatte haben.

So, wie ich es im Beispiel mit dem Zugriff auf die gesamte Festplatte gezeigt habe, kann man es ebenso mit Ordnern bis zu einzelnen Dateien

Bild 17: Nach der Rechteübernahme



durchführen und dies auf relativ einfachem Weg.

Sinnvoll ist es aber, sich ein Konzept zu überlegen, die Rechte auch sinnvoll zu setzen. Dabei kann eine durchdachte Ordnerstruktur nur helfen.

Man kann selbst regeln, was man in einem Ordner machen darf. Darf man in diesem Ordner Dateien hinzufügen und ändern *(Ändern)*, darf man enthaltene Dateien lesen und Programme ausführen, aber keine Dateien hinzufügen oder ändern *(Lesen, Ausführen)* oder darf man nur den Ordnerinhalt anzeigen und weder Dateien öffnen noch löschen *(Lesen, Ordnerinhalt anzeigen)*. Und diese Struktur kann man auch ruhig mal auf einem Blatt Papier festhalten, das ist keine Schande, das machen selbst Profis.

Wichtig ist es auf jeden Fall, von oben nach unten mit den Rechten zu beginnen, d. h. von der gesamten Festplatte hin zu den einzelnen Ordnern und bei Bedarf zu einzelnen Dateien. Hier wird man vom Betriebssystem auch nach der Vergabe der Rechte unterstützt, wenn zum Beispiel neue Unterordner angelegt werden, dann erbt dieser Unterordner die Rechte vom darüber liegenden Ordner. Dies kann man aber im Nachhinein ebenso noch ändern (Vererbungskette unterbrechen), wenn dies erwünscht ist.

Unter XP wurde der User meist nicht damit konfrontiert, weil er sowieso immer als *„Administrator"* gearbeitet hat.

Es gab zwar unter XP auch schon eine Rechteverwaltung, die aber im Privatbereich nie wirklich praktikabel war. Zum Beispiel wenn man einen Virenscanner unter einem Konto installierte, war dieser im anderen Konto nur eingeschränkt benutzbar. Das war nicht unbedingt immer die „Schuld" von Microsoft, sondern auch der Programmierer, die externe Software nicht sauber programmiert haben.

In dieser Hinsicht hat die UAC auch eine gewisse Hürde geschaffen, die der Verbesserung der angewendeten Programme nur zuträglich sein kann. Heutzutage gibt es jede Menge Programme, die keiner Zustimmung der UAC bedürfen. Das ist immer ein gutes Zeichen. Sobald etwas sich tief ins System verkrallen will, poppt die UAC auf und fragt nach einer Bestätigung.

Du kannst es auch so ausdrücken: Sobald die UAC aufpoppt, hast Du den Status (und das Wissen) eines Administrators nötig. 98 % der User sollten eigentlich nie eine UAC sehen. Wenn ich „normal" arbeite, werde ich nie mit Systemsachen konfrontiert. Soweit die Theorie.

Da aber zum Beispiel alle installierte Software periodisch nach Updates verlangt, holt sich die UAC dadurch vom User die Berechtigung > sprich Du gibst mit einem Klick Dein Okay.

Mit so was müssen wir leben. Da Du ja im Privatbereich gleichzeitig User und Administrator bist, hast Du auch die volle Verantwortung für Dein System.

FAT32…

FAT32 *(File Allocation Table)* ist das bevorzugte Dateisystem für USB-Sticks. Warum? Weil mit Fat32 jedes Betriebssystem (Windows, Linux, Apple…) problemlos umgehen kann. (da FAT32 keine Rechtestruktur besitzt).

Obwohl viele Betriebssysteme schon ganz gut mit NTFS umgehen können, einigt man sich immer auf den kleinsten gemeinsamen Nenner. Und der ist seit gut 20 Jahren FAT32.

Das FAT Dateisystem bietet ganz einfach nicht die Möglichkeit, Zugriffsrechte zu verteilen also zum Beispiel verschiedenen Benutzern unterschiedliche Zugriffsrechte auf Ordner oder Dateien zu geben.

Mit FAT32 kannst Du nur Freigaberechte auf Ordnern vergeben - hier sind die Berechtigungen auf *Lesen, Schreiben und Vollzugriff* beschränkt und wirken sich auf alle Dateien und Unterordner gleichermaßen aus.

Prinzipiell kannst Du das zwar auch für Gruppen und Benutzer unterschiedlich konfigurieren - aber eben nur mit den drei Auswahlmöglichkeiten. Ist nicht sehr praktikabel…macht man auch nicht so…

Mit NTFS kannst Du neben den Freigabeberechtigungen auch für jede Datei und jeden Ordner Zugriffsrechte vergeben (zum Beispiel Lesen, Schreiben - aber nicht löschen). Das kann für jeden Unterordner oder jede Datei unterschiedlich definiert werden.

Für jeden Benutzer oder Gruppe sind die Berechtigungen im unteren Bereich aufgelistet. Wenn man die NTFS mit den Freigabeberechtigungen von FAT32 vergleicht, stellt man fest, dass es hier (bei NTFS) wesentlich mehr Berechtigungen gibt.

Also, wenn Du lokal auf Windows was mit Berechtigungen zu tun hast, sind das immer NTFS-Berechtigungen.

Der Administrator…

Als Administrator bist Du *„Herr über alle Dateien auf deinem Rechner".* Stimmt. Nur erwartet Windows 7 schon, dass Du Dich an die prinzipiellen Vorgaben hältst und einige Dinge beachtest.

Du musst zum Beispiel sicher sein, dass im betreffenden Dateisystemobjekt ein Benutzer aus der Gruppe der Administratoren eingetragen ist und Berechtigungen besitzt.

Fehlt zum Beispiel der Benutzer aus der Gruppe der Administratoren, „weiß Windows" nichts davon und vergibt auch keine Zugriffsrechte. Dies ist u. U. der Fall, wenn Dateien von Windows XP-Datenträgern plötzlich nicht mehr zugreifbar sind. Siehe oben das Beispiel mit dem externen Datenträger.

Du kannst als Standardbenutzer und als Administrator jederzeit eine Datei per Rechtsklick anwählen und *„Eigenschaften"* wählen. Auf der Registerkarte *„Sicherheit"* findest Du die eingetragenen Benutzer, die überhaupt Zugriff auf die Datei haben. Über die Schaltflächen der Registerkarte lassen sich Berechtigungen anpassen oder hinzufügen. Die Schaltflächen sind durch ein stilisiertes „Schild" (blau-gelb) gekennzeichnet, als Hinweis, dass dies administrative Berechtigungen erfordert.

Fehlt dem Benutzer, der diese Änderungen durchführen soll, das entsprechende Sicherheitstoken, ruft die betreffende Funktion die Benutzerkontensteuerung auf. Wird diese vom Benutzer bestätigt, erhält der Prozess das Token bzw. wird auf das benötigte Sicherherheitslevel gehoben (läuft ab da mit Administratorrechten). Unter einem Administratorkonto kann es aber auch sein, dass die Sicherheitsabfrage nicht mehr erscheint, weil Microsoft da optimiert hat und die Berechtigungen automatisch erteilt hat. Habe ich noch nicht wirklich ein Muster erkennen können…

Was mir noch aufgefallen ist…

Unterschied zu XP…

Man kann in Windows7 nicht mehreren Dateien auf einmal neue Rechte zuweisen. Das heißt auf Deutsch, ich kann nur ordnerweise die Berechtigungen setzen, oder eben für jede Datei einzeln… sehr mühsam das Ganze. Das ging doch unter XP noch tadellos, dass man mehrere Dateien markierte und neu berechtigen konnte. Geht in Windows7 nicht mehr. Ärgerlich.

Workaround

Die zu ändernden Dateien in ein temporäres Verzeichnis auf der gleichen Partition verschieben. Im temporären Verzeichnis die Rechte anpassen und an die untergeordneten Objekte vererben. (anhakerln)

Dann die Dateien wieder an ihren ursprünglichen Ort verschieben.

Wichtig bei sämtlichen Aktionen ist nur, dass Du immer verschiebst. Und immer innerhalb derselben Partition.

Warum? Beim Verschieben innerhalb einer Partition behält die Datei ihre Berechtigungen. Beim Kopieren hingegen erbt sie die Berechtigungen des Zielordners. Also > immer verschieben.

Ist zwar nicht elegant, funktioniert aber.

Windows 7 Mail…

Dann gibt's da noch die leidige Sache mit dem Mailprogramm. Standardmäßig ist ja keines drin bei Windows7 und Microsoft empfiehlt Windows-Live-Mail. Ist aber nicht jedermanns Sache und außerdem hat es einen nervenden Penetrationsfaktor mit seinem *„Windows Live Account",* den man tunlichst einrichten sollte > rechts oben der Button *„Anmelden".*

Das Irre ist ja, dass ein „fast" fertiges Mail-Programm in Windows 7 enthalten ist. Das von Vista her bekannte *„Windows Mail"* im Ordner `„C:\Programme\Windows Mail\".`

Der Haken dabei: Das Programm funktioniert nicht. Grund ist die Datei `„msoe.dll"` im Ordner `„C:\Programme\Windows Mail\".` Microsoft hat diese Datei…hm ich sag mal verändert, oder unbrauchbar gemacht. Auf jeden Fall funktioniert sie nicht mehr.

Warum das Microsoft so gehandhabt hat? Vielleicht wollte MS ein EU-konformes Betriebssystem ohne ein aufoktroyiertes Mailprogramm auf den Markt bringen. Damit das Mailprogramm nicht funktioniert, wurde eben die `„msoe.dll"` manipuliert. Warum ein nicht funktionierendes Mail-Programm da im System defaultmäßig überhaupt vorhanden ist… no Ahnung?

Selbst meine diesbezügliche Anfragemail an „stif.balma@maikrosoft.uesa" wurde bisher vom Empfänger erfolgreich negiert. Obwohl ich sie extra mit einem „Fähnchen" als dringend gekennzeichnet habe…

Wenn Du also die Datei `„msoe.dll"` im Ordner `„C:\Programme\Windows Mail\"` durch eine Version von Vista ersetzt, läuft „Windows Mail" tadellos.

Trusted Installer…

Du musst aber einiges beachten, wenn Du die `msoe.dll` ersetzen willst: Hier kommt jetzt der Begriff „TrustedInstaller" ins Spiel.

Da diese Datei (`msoe.dll`) dem „TrustedInstaller" gehört (TrustedInstaller ist der Besitzer), kann man sie nicht einfach durch eine Vista -Version ersetzen.

„warum gät des net, i bin jo Admin", tönt es dann meist von den billigen Plätzen Richtung Monitor.

Erst muss man die Besitzrechte übernehmen und anschließend kann man sich die Rechte der

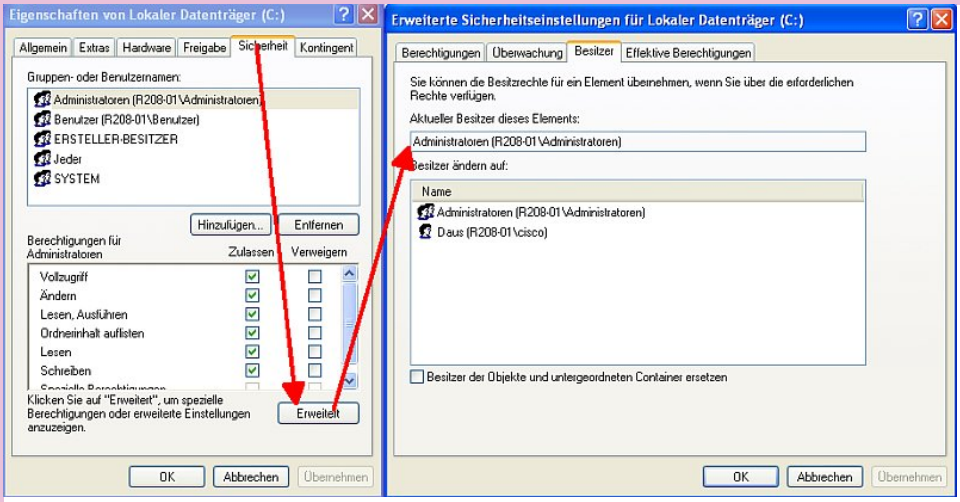


Bild 18: Trusted Installer durch Benutzer ersetzen

Datei aneignen. Die Rechteverwaltung funktioniert immer nach diesem „einfachen“ Schema.

Danach ersetzt Du die Datei (`msoe.dll`) durch eine Vista-Version. Fertig.

Der Administrator hat einfach die Macht, sich überall Zugang zu verschaffen.

Er hat aber nicht überall am System defaultmäßig Zugang.

Es ist deshalb auch ganz im Sinne von Microsoft, dass bei Systemfiles keiner Zugriff haben soll. Außer eben der TrustedInstaller. Das ist ein Sicherheitsfeature, nichts anderes.

Zur Ergänzung > In Linux hast Du als „root“ überall am System Zugang…

Trusted Installer = *Vertrauenswürdiger wos a immer*

Der TrustedInstaller wurde beim Setup angelegt. Das ist auch der Grund, dass das meiste Zeug in „C“ den TrustedInstaller als Besitzer hat. Da haben selbst Administratoren keinen Zugriff drauf. Auf anderen Partitionen (D, E…) ist meist der User SYSTEM der Besitzer der Objekte. Das hat schon seine Richtigkeit. Lass es so, außer Du weißt was Du tust…

Hört man öfter, dass man trotz Adminrechten keinen Zugriff auf „C“ hat.

Wie funktioniert das mit dem TrustedInstaller und wie erlange ich trotzdem Zugriff auf „C“?

Natürlich gibt es Situationen, wo man an die Systemfiles ran muss. Wie machst Du das? Einfach weiterlesen…

Eine Warnung noch: Natürlich kann man sich die Dateien unter „C“ aneignen. Empfehlen tu ich es explizit nicht. Jede Manipulation am Laufwerk „C“ kann die Systemsicherheit beeinträchtigen.

Des weiteren sollte nie die komplette „C“ Partition „übernommen“ werden.

Ich habe Dich gewarnt…

Gehen wir‘s trotzdem durch:

Wie? So: *Arbeitsplatz -> Laufwerk C: -> Rechtsklick -> Eigenschaften*. Dort klickst Du unter der Karteikarte *„Sicherheit“* auf „Erweitert“. Letzter Schritt wäre, dass Du bei *„Besitzer“* den TrustedInstaller durch deinen Benutzernamen änderst. **Bild 18**

Ist zwar ein Screenshot von einem XP, funktioniert aber in Windows 7 genauso.

Dann schaust Du im Task Manager nach, ob unter *„Dienste“* die „`trustedinstaller.exe`" gestartet ist oder der Dienst beendet wurde. Auf keinen Fall solltest Du diesen Dienst beenden, da sonst Schwierigkeiten mit der Updatefunktion von Windows zu erwarten sind. Und was weiß ich noch…

Der TrustedInstaller verhindert im Grundlegenden das „heimliche“ oder unabsichtliche Verändern von einzelnen Dateien; insbesondere der Systemdateien. Ist also ein Sicherheitsfeature, das natürlich in gewissen Situationen auch lästig sein kann.

Die Dateirechte an einer originalen Systemdatei weisen dem Benutzerkonto, Systemkonto und den Administratoren lediglich Lese- und Ausführen-Rechte zu. Vollzugriff hat nur eine Benutzergruppe namens TrustedInstaller, die im Besitz aller Systemdateien ist.

Selbst der Default-Administrator darf also nicht alles - weder eine Systemdatei umbenennen, ändern noch löschen. Der Dienst steht im Taskmanager unter *„Dienste“*, wenn er aktiv ist (was er eigentlich immer sein sollte), aber nicht unter *„Dienste“* (`services.msc`) in der Verwaltung oder unter Benutzerkonten.

Müssen gezwungenermaßen Systemdateien verändert werden, gibt es aber doch eine Möglichkeit.

Um die Datei zugänglich zu machen, müssen die Sicherheitseinstellungen des Ordners der betreffenden Datei abgeändert werden.

Gilt für den Administrator:

1. Die *„Eigenschaften“* des Ordners aufrufen - Registerkarte *„Sicherheit“* - Button *„Erweitert“* - Registerkarte *„Besitzer“* - Aktueller Besitzer ist TrustedInstaller - Button *„Bearbeiten“*. Besitzer ändern nach *„Administratoren“* anklicken und Button *„Übernehmen“*.

   Pass auf, dass Du die Gruppe der Administratoren auswählst > Symbol mit 2 Köpfen.

2. Windows-Sicherheitsdialog bestätigen und dann noch *„OK“*.

3. Aktueller Besitzer sollte nun die Gruppe der *„Administratoren“* sein. Also, den Besitz hätten wir schon mal. Jetzt noch die Berechtigungen vergeben.

4. Gehe auf die Registerkarte *„Berechtigungen“* – *„Administratoren“* anklicken - Button *„Bearbeiten“* - Gruppen- oder Benutzernamen – *„Administratoren“* anklicken - Button *„Bearbeiten“*

5. *„Administratoren“* anklicken und unter *„Berechtigungen für Administratoren“* das Häkchen unter Zulassen setzen – *OK*.

6. Sollte passen.

   Jetzt für den Benutzer, dieselbe Vorgehensweise wie für den *„Administrator“*.

   Zum Schluss aber *„Benutzer“* anklicken - Button *„Bearbeiten“* – *„Zulassen“* - *OK* – *„Übernehmen“*.

7. Windows-Sicherheitsdialog inbrünstig bejahen

8. *„Fehler beim Anwenden der Sicherheit“* Zugriff verweigert - Button *„Fortsetzen“* klicken. Was bleibt uns über.

Nun sollten die Berechtigungen von *„Administratoren“* und *„Benutzer“* statt *„TrustedInstaller“* alle Rechte haben. Der Zugriff auf die Systemdatei sollte nun klappen. Alternativ würde ich aber vorher mit einem anderen Dateimanager (a43, Explorer++...) das Ganze versuchen. Ist vielleicht stressfreier.

Noch ein Wort zu den Vererbungen…

Hier gibt es erfahrungsgemäß auch immer wieder Probleme.

Wie Du den Besitz von Objekten übernehmen kannst, habe ich ja oben eh schon angerissen. Sollte funktionieren. Wenn Du danach die Rechte vergibst, solltest Du aber auch aufpassen, was Du machst.

Gehen wir‘s durch:

Du gehst auf einen Ordner mit Rechtsklick > *Eigenschhaften* > *Sicherheit*...kennen wir ja schon.

Unten auf den Button *„erweitert“*.

Dann siehst Du so was in der Art: **Bild 19**

In diesem Fenster gehst Du zur Registerkarte *Berechtigungen* und klickst unten auf den Button *„Berechtigungen ändern“*.

Dort entfernst Du zunächst das Häkchen vor *„Vererbbare Berechtigungen des übergeordneten Objektes einschließen“*, um zu verhindern, dass deine manuellen Änderungen durch automatische Weitergabe von Rechten aus darüber liegenden Verzeichnissen wieder überschrieben werden. **Bild 20**

Das heißt einfach, dass Du die Vererbungskette unterbrichst. Darum ist es auch so wichtig, dass Du bei den Rechten immer von oben nach unten arbeitest. **Bild 20**

Du wirst nun gefragt, ob Du die bestehenden geerbten Berechtigungen hinzufügen oder entfernen möchtest: **Bild 21**

Hinzufügen bedeutet, dass die Berechtigungen zunächst unverändert bleiben, aber nicht mehr geerbt werden. Das wollen wir. Drück‘ da drauf.

Wenn Du auf *Entfernen* klicken solltest, werden alle existierenden Berechtigungen gelöscht. Ich hoffe, Du weißt bei dieser Option, was Du tust.

Bei Verzeichnissen aktiviere noch unten die Option *„Alle Berechtigungen für untergeordnete Objekte durch vererbbare Berechtigungen von diesem Objekt ersetzen“*. **Bild 22**

Das stellt sicher, dass die Änderungen, welche Du nun vornimmst, auf alle Dateien und Unterverzeichnisse dieses Ordners angewendet werden.

Hinweis: Bei einzelnen Dateien wird diese Option nicht angezeigt. Logo…

Klicke nun auf die Schaltfläche *Hinzufügen* und gib deinen Benutzernamen ein, mit dem Du angemeldet bist. Danach klickst Du auf die Schaltfläche *Namen überprüfen*, um Tippfehler auszuschließen. Dein Name erscheint dann unterstrichen. Das bedeutet, dass er vom System erkannt wurde.

Die Maske dürfte uns auch nicht unbekannt sein> **Bild 23**

Anschließend mit *OK* bestätigen.

Klicke auf die oberste Option *Vollzugriff*, und lass die übrigen Einstellungen unverändert. **Bild 24**

Schließe das Fenster, indem Du auf *OK* klickst.

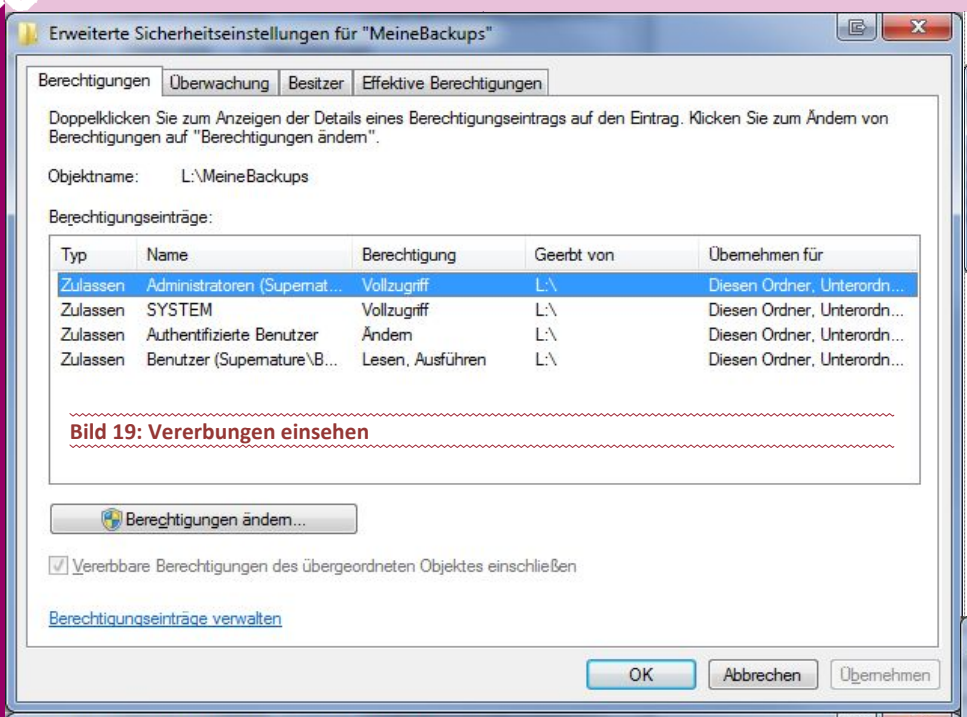


Bild 19: Vererbungen einsehen

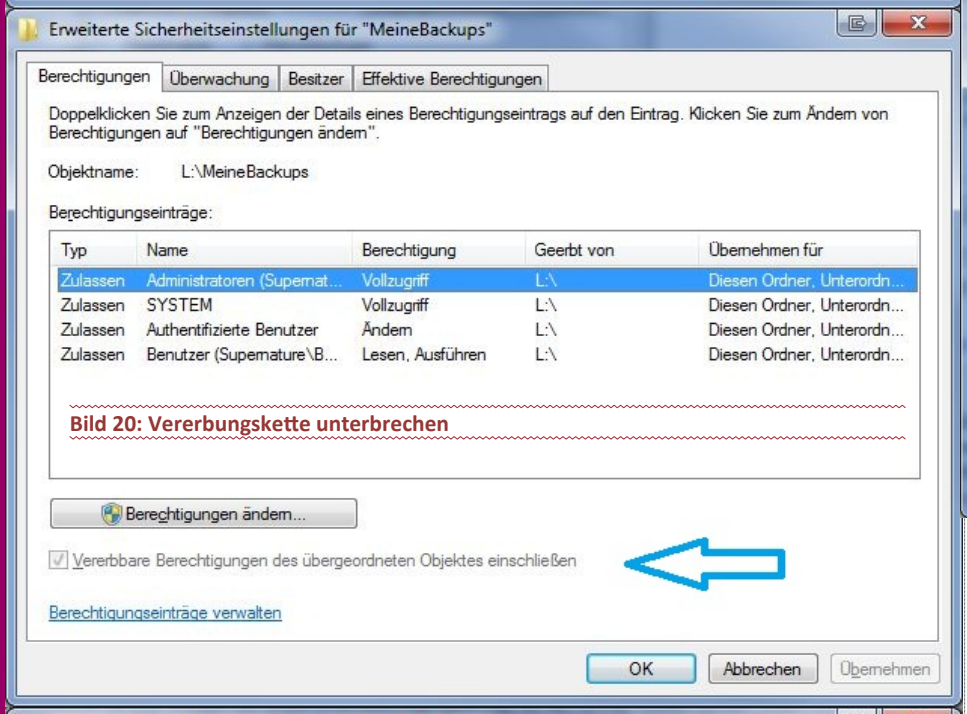


Bild 20: Vererbungskette unterbrechen

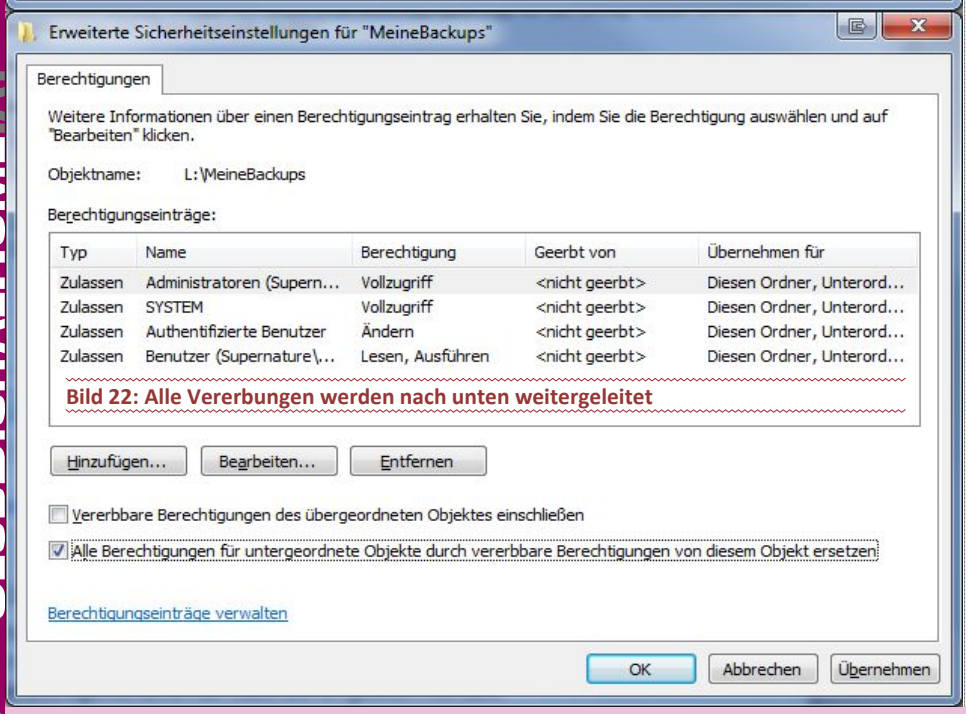


Bild 22: Alle Vererbungen werden nach unten weitergeleitet

Bild 21: Wie werden Vererbungen behandelt

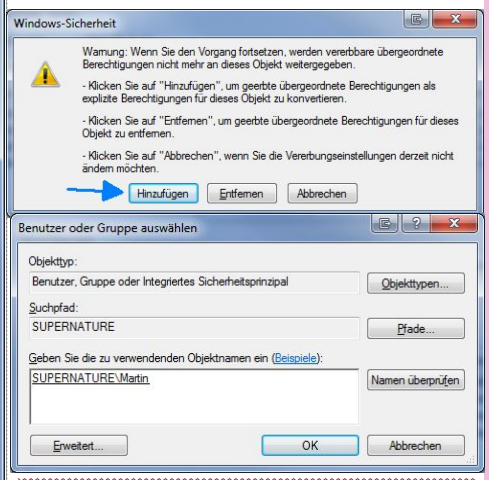


Bild 23: Unterstrichene Namen wurden vom System erkannt

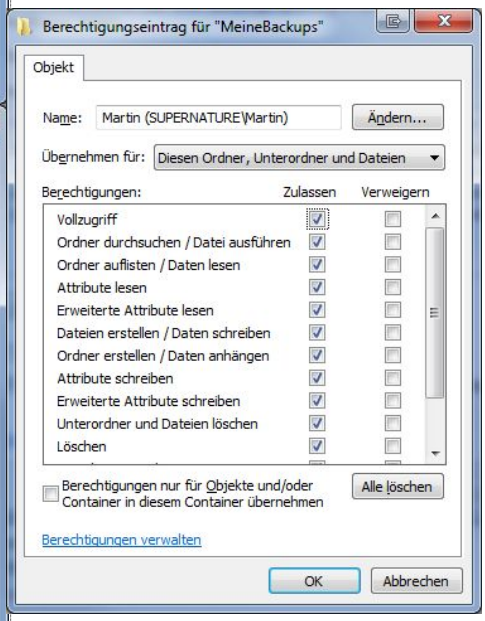


Bild 24: Granulare Rechtezuweisung

Im nächsten Fenster (wir sind wieder auf **Bild 22**) *„Erweiterte Sicherheitseinstellungen"* gehst Du noch auf *OK*, um die neuen Berechtigungen zu übernehmen.

Das kann jetzt ein bisschen dauern. Mach inzwischen ein Delirium-verheißendes Destillat (zum Beispiel Bier) auf und verkappe es oral. Das hast Du Dir jetzt verdient. Wenn der ganze Zauber mit den Berechtigungen durchgerannt ist, sollte es fertig sein.

Wahnsinn, jetzt hab ich Dich mit dem Windows-Zeug zugelabert. Und wir sind erst bei der Hälfte! Jetzt kommt auch noch die Linux-Seite dazu…puh…das wird wirklich ein langer Artikel.

Linux

Die Linux-Rechteverwaltung ist mit der von Windows nicht vergleichbar.

All Dein Windows-Wissen ist jetzt eher hinderlich. Ich weiß, dass man anfangs immer versucht, gewisse Parallelen zu finden und anzuwenden. Vergiss das. Es kostet Dich nur mehr Zeit und unnötige Nerven.

Noch einmal: Gehe in Linux vollkommen unvoreingenommen an die Rechte-Thematik ran. Es ist alles ziemlich logisch aufgebaut.

Wenn Du allerdings nur User bist, und bei eventuellen Problemen sowieso kompetente Hilfe zur Hand hast, kannst Du dieses Kapitel auslassen.

Die Rechteverwaltung in Linux ist meines Erachtens um einiges einfacher zu handhaben als in Windows.

Ich weiß, das ist sehr subjektiv von mir. Der Vorteil für einen Linux-User liegt in der Nachhaltigkeit des Systems. Einmal begriffen, kannst Du das jahrelang anwenden.

Vor allem haben die Linux-User einen entscheidenden Vorteil gegenüber den Windows-Jüngern: Linux arbeitete immer schon mit einer (derselben) Rechteverwaltung. (schon Anfang der 90er Jahre, als die Windows-Privatuser noch mit Windows98 herumfuhrwerkten). Also *„dass man wo keinen Zutritt hat“*, dieses Szenario gab‘s in Linux schon immer.

Bei Windows musst Du mit der UAC hingegen wieder umlernen. Erschwerend kommt dort noch hinzu, dass Microsoft seine Betriebssysteme so aufbaut, dass auch Volltrotteln nicht zu viel kaputt machen können (sollten). Sorry für diese Wortwahl…

Daraus resultieren dann so ärgerliche aufpoppende Abfragedialoge wie *„wollen sie wirklich löschen…Gratuliere, sie haben soeben eine Sicherung angelegt…“*.

Solche Dialoge gibt’s bei Linux nicht. Dort kommt nur eine Meldung, wenn‘s einen Fehler im Ablauf gibt. Eine Fehlermeldung eben.

Das heißt, Du wirst in Linux nicht permanent mit Dialogboxen konfrontiert, die Du abnicken musst.

Linux Credo > *no news is good news*.

Den Spagat, den Microsoft da seit Jahren hinlegt, um Profis gleichermaßen wie blutige Anfänger zu bedienen, bringt eben gewisse Einschränkungen bezüglich des Systemzuganges mit sich.

Man kann es auch so ausdrücken:

Linux ist für den mündigen User gemacht. Zumindest war es immer schon so angedacht.

Windows (auch) für den DAU *(dümmster anzunehmender User)*.

Und verwechsle benutzerfreundlich nicht mit idiotenfreundlich… ganz wichtig.

Wenn‘st nicht nachdenkst, machst sowieso was hin. Egal, ob in Linux oder Windows.

Ich weiß, dass mir diese Zeilen wichtige Sympathiepunkte kosten werden. Da werde ich dann wohl mit meinem Therapeuten Rücksprache halten müssen… da muss ich durch.

Oder als Administrator hast Du nicht überall Zugriff in Windows (Systemfiles…)

In Linux darf *root* alles machen. Auch „unsinnige“ Dinge. Du bist *root*, Du musst wissen was Du tust. Keiner verlangt von Dir, dass Du als *root* arbeiten sollst. Wenn Du keine Ahnung von dem ganzen Zeug hast, lass‘ es.

Ich war mal in einem Linux-Kurs, wo ein Teilnehmer das komplette System und das zusätzlich eingehängte Windows XP mit einem Befehl gelöscht hat.

Übrig blieb eine leere NTFS-Partition und eine Linux-Partition mit ein paar unbrauchbaren Trümmern drauf. In 3 Sekunden war der Spuk vorbei. Keine Sicherheitsabfrage, keine aufpoppenden Warnboxen… nichts kam. Warum auch. *Du bist root, Du musst wis… eh scho wissen :-))*

Das „Schöne“ an dem Szenario war ja, dass man dem Verursacher richtig ansah, wie es vom Hals aufwärts schön langsam warm wurde… Es dauerte natürlich ein bisschen, bis er sich der kompletten Tragweite des verhängnisvollen Befehles bewusst wurde. Man will es anfangs nicht glauben. Ein Neustart des Systems zeigte aber lediglich *„Kein Bootmedium gefunden…“* an. Die Platte war leer…

Gewiss war nur eines: Der Befehl funktioniert.

Deshalb legt man in Linux auch immer einen „normalen“ Benutzer an, der keinen Zugriff auf Systemsachen hat. Falls man diese braucht, wechselt man vom normalen Benutzerkonto mittels su in den erhöhten Level.

su = *switch user* > Identität von anderem User annehmen (üblicherweise *root*)

Rechte in Linux

Kommen wir zu den Rechten:

Wenn ein erfahrener Windows-Admin das erste Mal mit Linux-Rechten in Kontakt kommt, ist er grundsätzlich mal enttäuscht. Anders kann man es nicht beschreiben.

Die Windows-Rechteverwaltung ist sehr komplex. Um nicht zu sagen kompliziert.

Für die Windows-Rechteverwaltung hast Du üblicherweise einen 100-seitigen Wälzer durchzuackern, um halbwegs den Durchblick zu bekommen (Benutzer, NTFS-Rechte, Vererbungen, Gruppen, Administratoren, UAC, Systembenutzer, Profile...). Netzwerk- und Domänenszenarien sind da überhaupt noch nicht eingerechnet. Es geht nur um den Heimbereich.

In Linux reichen in der Regel fünf A4-Blätter, um die Rechteverwaltung zu verinnerlichen. (user, root, Gruppen, chmod, chown, ...)

Die Linux-Rechteverwaltung ist sehr einfach gestrickt. Ist so. Im Unternehmensbereich ist es aber bei beiden Systemen ziemlich komplex.

Wir bleiben aber erst einmal im Heimuserbereich.

Beide Rechteverwaltungen (Windows + Linux) funktionieren sehr gut. Wenn man weiß, was man macht.

Da die meisten Privatuser sich deren Handlungen in diesem Bereich gar nicht bewusst sind, gibt’s öfter Brösel.

Ehrlicherweise muss ich sagen: Deutlich mehr in Windows als in Linux. Warum?

Die Komplexität der Windows-Rechteverwaltung hätte schon vor 11 Jahren (W2k) vielen Privatusern zumindest bekannt sein sollen. Sie wurde leider nie (oder selten) im Privatbereich angewendet. Ergänzend muss man aber auch sagen, dass bis XP ein Mehrbenutzerbetrieb extrem mühsam war, oder besser gesagt nicht praktikabel.

Bis XP konnte man das Ganze noch elegant standardmäßig umgehen. Man war sowieso als *„Administrator“* immer angemeldet und konnte auf der Kiste machen, was man wollte.

Seit Vista und Windows 7 gibt‘s das in dieser Form nicht mehr. Die User werden automatisch mit der Rechteverwaltung konfrontiert. Ob sie wollen oder nicht.

Eine Rechteverwaltung impliziert auch immer gewisse Einschränkungen. Ist so.

Das heißt, das Gros der Windows-User ist heutzutage (2011) zum ersten Mal in ihrem Leben mit Dingen konfrontiert, die für sie am PC *„verboten“* sind. *„…wieso derf I do nix mochn… der losst mi net… kann nix kopieren… da PC ghert jo mir… usw“*.

Ein komplettes Neuland eben. Das Windows98 Feeling konnte noch relativ leicht nach XP hinübergerettet werden. Danach war aber mit Vista (und der aufoktroyierten Rechteverwaltung) schon Schluss.

Microsoft „zwang“ somit seine User, umzudenken. Oder Du bleibst bei XP.

Daraus lässt sich schon mal einer der größten Vorteile von Linux herauskristallisieren.

Linux hat knapp 20 Jahre Vorsprung in der User-Akzeptanz einer Rechteverwaltung gegenüber Windows. Wenn‘st die Unix-Zeit auch einbeziehst (Linux stammt von Unix ab), sind‘s sogar gut 40 Jahre.

Soll heißen, die Rechteverwaltung in Linux war immer schon ein fixer Bestandteil. Weder abschaltbar oder granular (per UAC) irgendwie einstellbar. Auf jeder Sche*ss Linux-Box läuft dasselbe Rechtesystem. Ohne Rechtesystem würde Linux auch gar nicht funktionieren. Darum wird auch nie von der Userseite angedacht, dieses zu manipulieren, sodass *„man alles machen kann“*.

Auf Windows-Seite müssen die User sich erst mit der Rechteverwaltung anfreunden. Das muss man einem User einmal verklickern, dass er nicht alles am System machen darf.

Die Akzeptanz in dieser Hinsicht gibt‘s meiner Erfahrung nach meist bei Profis und mündigen Usern.

Schätzungsweise 40 % manipulieren die UAC meiner Meinung nach. Warum auch immer.

Kurz gefasst:

Linux > Die Rechteverwaltung ist (und war) immer schon in Linux verankert.

Windows > Rechteverwaltung ist seit Vista (2007) Standard.

Genau genommen kann Linux Zugriffsrechte nur für Dateien vergeben. Das Gute daran ist, dass alles unter Linux als Datei behandelt wird. (CD-ROM-Laufwerk, Drucker, Prozesse und andere Hardware…). Auch ein Ordner ist nichts anderes als eine (spezielle) Datei.

Eine Datei ist einem Besitzer und einer Gruppe zugeordnet. So kann Linux den Dateizugriff in drei Stufen regeln: für den Besitzer, für die Mitglieder der Gruppe und für die übrigen Benutzer.

Zugriff kann heißen: die Datei lesen, die Datei schreiben (anlegen, verändern, löschen), die Datei ausführen (ein Programm starten).

Programme laufen mit den Rechten des Benutzers, der sie gestartet hat. Getreu dem Linux-Grundsatz, dass kein Benutzer mehr Rechte bekommt, als er für seine Arbeit benötigt, richtet Linux für bestimmte Dienste spezielle virtuelle oder Systembenutzer mit sehr beschränkten Rechten ein.

Einen User anlegen… klassisch.

Im Zeitalter der GUIs legst Du neue User entweder grafisch per Dialogboxen oder klassisch auf der Shell mit dem Befehl „`adduser`" an.

Wie sieht das aus? So, is in spanisch, aber man weiß trotzdem um was es geht. Die einzigen 2 Befehle, die ich eingegeben habe, wurden fett hervorgehoben. Alles andere wurde vom System als Bestätigung ausgegeben (außer meine 2malige Eingabe des Passwortes)

Nach dem > sind meine Kommentare dazu, was gerade passiert…

```
wachbirn@antichrist:/home/wachbirn$ su
```
> ich wechsle auf „root“

```
contrasena:
```
> root-Passwort eingeben

```
root@antichrist#
```
> Du siehst, das Dollarzeichen wurde gegen eine Raute getauscht…und natürlich auch der username gegen „`root`"

```
root@antichrist# adduser testuser
```
> erzeuge user „testuser“

Añadiendo el usuario `testuser'… > user „testuser“ wurde hinzugefügt

Añadiendo el nuevo grupo `testuser' (1001) > Gruppe „testuser“ hinzugefügt

Añadiendo el nuevo usuario `testuser' (1001) con grupo `testuser'> alles bestätigt

Creando el directorio personal `/home/testuser'…> home verzeichnis erzeugt

Copiando los ficheros desde `/etc/skel'…> standardeinstellungen für neuen User in home-Verzeichnis hineinkopiert

Introduzca la nueva contraseña de UNIX: > passwort für neuen User

Vuelva a escribir la nueva contraseña de UNIX:> noch mal Passwort

passwd: contraseña actualizada correctamente > Passwort wurde 2 mal korrekt eingegeben…

Cambiando la información de usuario para testuser> Zusatzinformationen… Zimmer, Telefon…wers braucht halt…

Introduzca el nuevo valor, o presione ENTER para el predeterminado

Nombre completo []: kompletter Name

Número de habitación []: Wohnadresse

Teléfono del trabajo []: Telefon Arbeit

Teléfono de casa []: Telefon zu Hause

Otro []: Andere Nozizen…

¿Es correcta la información? [S/n]> Enter drücken und der User ist angelegt. Fertig.

Das Ganze hab ich wie bereits erwähnt, von meinem normalen Benutzerkonto aus eingerichtet. Mit dem Kommando „su“ (switch user) teilte ich dem System lediglich mit, dass ich eine andere Identität haben will. Das root-Passwort berechtigte mich sodann, Administratoraufgaben durchzuführen. Die visuelle Erkennung dafür war auch die Raute (#) am Zeilenende

selbstverständlich hätte ich auch eine andere Identität annehmen können. Wie? So:

```
wachbirn@antichrist$ su anderer_username
```

contrasena > natürlich geht das nur, wennst das Passwort von dem anderen User weißt.

Als *root* brauchst Du natürlich kein anderes Passwort eines Users zu wissen. Du hast überall Zugriff.

Die Voreinstellungen für jeden User stehen in der Datei /etc/default/useradd. **Siehe Bild 31.**

Hier sieht man recht schön, dass die angelegte Hauptgruppe immer die group-id 100 bekommt, und das Home-Verzeichnis unter „/home" erreichbar ist. INAKTIVE und EXPIRE sind 2 Variablen die nicht aktiv sind (-1 impliziert das). Standardshell ist „sh". sh ist ein Link auf bash. Und sämtliche Standardeinstellungen wurden vom Ordner „skel" übernommen. „skel" bedeutet *skeleton*=Skelett. Ist also die Grundausstattung jedes Users .

„adduser" ist lediglich ein Skript, das auf „useradd" verweist. Adduser ist einfach bequemer, da man nicht so viel tippen braucht.

Shell ist wie erwähnt die bash *(Bourne Again shell)*, Gruppe ist die selbe wie der Username (muss nicht überall so sein, aber meistens).

Kennziffer uid hat eine Zahl größer als 999. Sprich, ab der uid 1000 fangen die „wirklichen“ User an.

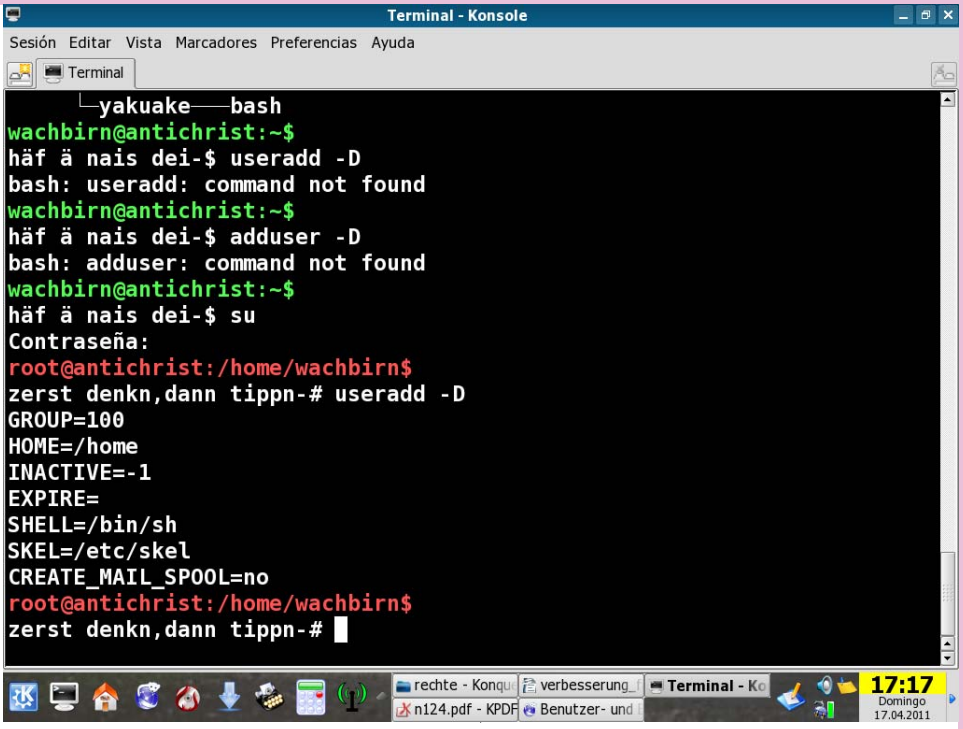


Bild 31: Useradd Informationen

Kleinere Zahlen sind für die Systembenutzer reserviert, *root* hat die Null. Du kannst mehrere Gruppen angeben, jeweils durch ein Komma getrennt. Damit der neue Benutzer mit der Arbeit beginnen kann, übernimmt Linux beim Anlegen aus dem Verzeichnis /etc/skel eine minimale Ausstattung für das Home-Verzeichnis mit Konfigurationsdateien für Shell und Window -Manager.

skel steht für Skelett= Grundgerüst für jeden neuen User.

Die Liste der Benutzer auf deinem System liefert die Datei /etc/passwd (passwd = Passwort-Datei). Guckst Du hier…mein Ausschnitt der Datei „passwd"

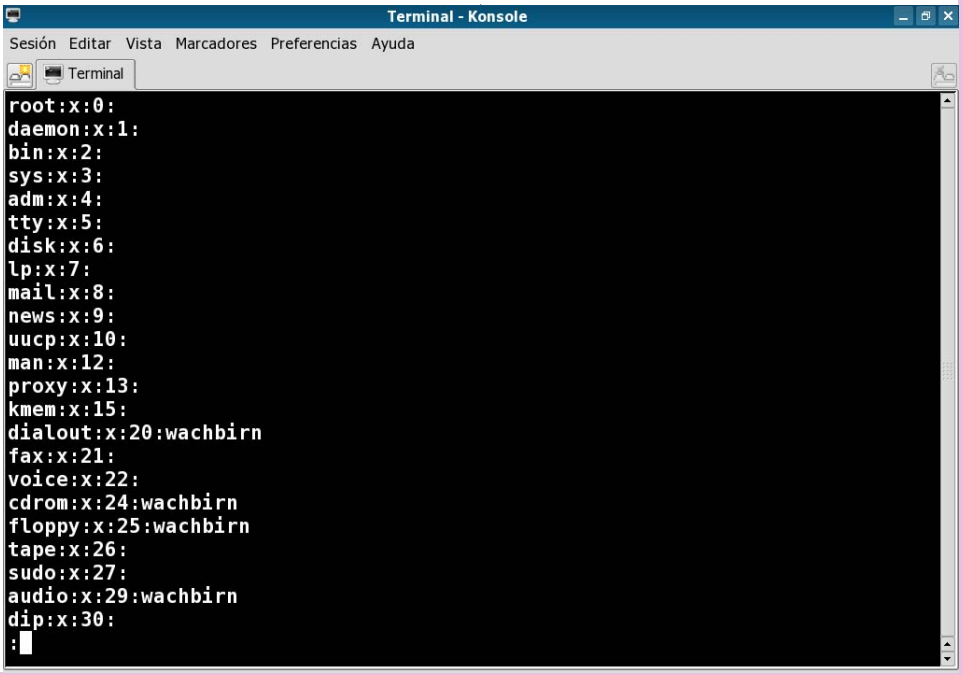
```
messagebus:x:104:107::/var/run/dbus:/bin/false

wachbirn:x:1000:1000:wachbirn,,,:/home/wachbirn:/bin/bash

avahi:x:105:109:Avahi mDNS daemon,,,:/var/run/avahi-daemon:/bin/false

festival:x:106:29::/home/festival:/bin/false

haldaemon:x:107:111:Hardware abstraction layer,,,:/var/run/hal:/bin/false

saned:x:108:114::/home/saned:/bin/false

testuser:x:1001:1001:,,,:/home/testuser:/bin/bash
```

testuser und wachbirn sind die einzigen zwei angelegten User auf meinem System. Alles andere sind „Systemuser“.

Das x nach dem Usernamen bedeutet, dass es ein Passwort gibt. Ohne x kann man sich eh nicht anmelden. Das Passwort steht aber immer in einer anderen Datei. Nämlich „shadow“.

Erklär‘ ich ein anderes Mal genauer. Muss schauen, dass ich da mit den Rechten weiterkomme…

Bild 32: Gruppen in Linux



Wer darf was…

Wer was unter Linux darf, steuerst Du über Zugriffsrechte auf Dateien. Da unter Linux auch die Hardware durch Dateien repräsentiert wird, kannst Du so Dein System aus einem Guss vollständig durchstrukturieren.

In Linux gibt's einen (oder mehrere) User, der natürlich keine Administratorrechte besitzt (auch keine eingeschränkten). Einen Administrator gibt's auf einem Linux-System immer. Das war's. Es gibt die User, und eben den Administrator. Der heißt immer *root* in Linux. Selbstverständlich gibt es auch Gruppen. Ist aber recht einfach gehalten. **Sieh auf das Bild 32.**

Alle Geräte werden durch Gruppen repräsentiert. Wie erwähnt. In Linux wird alles als Datei behandelt. Wenn Du nicht Mitglied einer Gruppe bist, darfst Du auch nicht dessen Features in Anspruch nehmen. Links stehen die Gruppen und nach dem Doppelpunkt die Benutzer, die darauf zugreifen dürfen. Wenn Du jetzt zum Beispiel unten bei der Gruppe „audio" den Benutzernamen „wachbirn" entfernst (einfach löschen aus dem file), so kann ich als Benutzer keine Musik mehr hören. *Simple as that.*

Viel mehr ist dazu nicht mehr zu sagen, da man das Ganze in der Praxis einmal nachvollziehen sollte. Ist aber keine Raketentechnik, ehrlich.

Wie funktioniert das? Ganz gut, würd ich sagen.

Darum gebe ich meist auch kein *root*-Passwort bei meinen Schäfchen her. Somit kann man schon mal den Zugriff jedweder Art ins System unterbinden.

Wenn ich als User erhöhte Rechte brauche, poppt eine Dialogbox auf, die mich nach dem Passwort des Admins fragt. Immer.

Im grafischen Modus poppt eine Dialogbox auf während auf der Konsole (wie Eingabeaufforderung in Windows, nur mächtiger) das Passwort abgefragt wird. **Bild 25**

Alle systemrelevanten Dinge in Linux kannst Du nur mit Kenntnis des root-Passwortes ändern. (Uhrzeit ändern, Software-und Druckerinstallation, editieren von Systemfiles, partitionieren…)

Du kannst aber auch nicht wie bei der Windows-UAC die Abfragelevel granular einstellen oder die Admin-Abfragebox gar deaktivieren. Das würde das Rechtesystem ad absurdum führen.

In Windows kannst Du hingegen das Rechtesystem einfach aushebeln. Wie? Deaktiviere die UAC. Ob das sinnvoll ist, ist eine andere Frage. Aber alleine schon die Möglichkeit der Deaktivierung ist so eine typische Microsoft-Lösung. A la wir sind auf dem Weg zu einem ausgereiften Usersystem… aber wir passen trotzdem auf, dass der User mit erhöhten Rechten nicht zu viel kaputt machen kann.

Mein Senf dazu: Sobald ich erhöhte Rechte habe oder brauche (egal ob in Windows oder Linux), muss ich wissen was ich tu.

Der Admin-Level ist eben nicht für den Normaluser gedacht. Das liegt in der Natur der Sache. Wie weit will Microsoft die Admins noch an der Hand führen? In diesem Level haben Gemüter mit schlichtem Geist sowieso nichts verloren. Diese eingebauten Kindersicherungen nerven mit der Zeit. Warum? Weil Du oft nicht direkt am System arbeiten kannst.

Sobald Du in Linux Root-Zugang hast, kannst Du alles machen… ohne Nachfragen, ohne Netz und dergleichen.

In Linux kannst Du als *root* alle Systemfiles löschen. (nein, die stellen sich nach dem Neustart nicht wieder automatisch her) oder die Systempartition formatieren. Ja, die Kiste ist nachher unbrauchbar, stimmt. Wem willst Du jetzt die Schuld geben?

„des hob I jo net gwusst…" gilt nicht. Du bist *root*. Du musst wissen, was Du tust.

Sieh es so: In Windows setzt Du Dich auf den Rücksitz und lässt Dich chauffieren. In Linux bekommst Du den Zündschlüssel in die Hand gedrückt. Wenn Du nicht fahren kannst, selber schuld. Dann nimm eben hinten Platz.

Keiner zwingt Dich dazu, als *root* (oder `Administrator` in Windows) zu arbeiten. Zur Erinnerung, es gibt Fachkräfte für so was. Die haben jahrelang gelernt, viele Kurse besucht und noch mehr Erfahrungen diesbezüglich gesammelt.

Auch wenn es Microsoft immer wieder versucht zu suggerieren. Nein, auch der Administrator in Windows setzt ein profundes Wissen voraus.

„`setup.exe`" doppelklicken, defragmentieren und `pagefile.sys`-Größe einstellen sind nach wie vor gute Einstiegspunkte ins Geschäft. Nicht mehr. Damit kann man vielleicht eine ahnungslose Hausmeisterin beeindrucken. Aber spätestens bei den Rechten musst Du nachdenken. Du musst auch dort genau wissen, was Du tust. Überlege lieber fünf Minuten, bevor Du nachher drei Stunden fluchst und letztendlich neu installierst.

Schon hundertmal erlebt, dass ein lieber Bekannter mit einer CD von einer Computerzeitschrift auftaucht und dann die empfohlenen Optimierungstools draufbügelt. Kann ja nicht schaden…

Somit schleppt die Kiste mit der Zeit immer mehr Zusatzsoftware mit, die der Übersicht, Stabilität und dem Ressourcenverbrauch nur abträglich sein kann.

Selber erlebt, wie in einem Kurs (kein PC-Kurs) ein Trainer demonstrativ den Registrycleaner (tuneup wars glaube ich) empfahl und sogleich auf seiner Kurskiste durchlaufen ließ. Tja, danach waren in der Registry die Office-Verknüpfungen verschwunden. Seufz…

Er bekam eh eine am Deckel vom herbeigeeilten EDVler. Egal…

Mein Tipp: Scheiß' auf die Optimierungstools.

Wenn Du Dein Windows optimieren willst, besuche einen Systemkurs oder eigne Dir Systemkenntnisse autoditakt an. Ja ich weiß, das kostet alles Zeit.

Bedenke, dass Du Dir Kenntnise eines Berufsbildes aneignen willst… (Systemadministrator, Netzwerktechniker…).

Wenn Du schweißen lernen willst, gehst Du auch erst ein paar Wochen in einen Kurs.

Mit der Zwei-Klick-Lösung der Optimierungstools wirst Du auf Dauer nicht glücklich werden.

Selbst für ein stinknormales „Registrycleaner-Zeugs" brauchst Du normalerweise ein paar Stunden Einarbeitungszeit, damit man einmal schnallt, was da überhaupt passiert. Auch wenn Du keine Zeit hast und die Kiste zum Arbeiten brauchst… wenn Du was am System machen willst, musst Du wissen was Du tust.

In meinen Augen sind die meisten Optimierungstools großartig… für den Windows-Supporter. Das gekoppelt mit der einhergehenden Rechteverwaltung sorgt im Privatbereich für stetige Support-Nachfrage.

An jeder Ecke bekommst Du im Internet alles Mögliche diesbezüglich aufgeschwatzt. Auch für Windows7. Leider…

Der philosophische Unterschied zu Linux?

In Linux hast Du schon alles an Bord, wenn Du was am System „optimieren" willst.

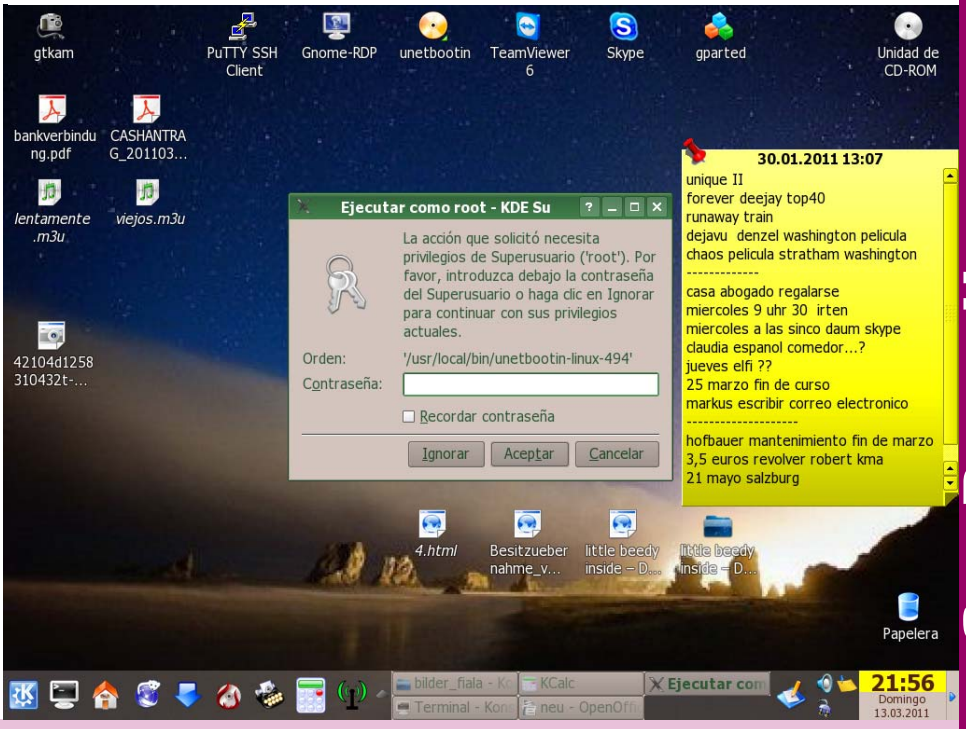
Ich hab noch nie eine Software im Internet gesehen, die Dein Linux um 30% schneller starten lässt. Wenn Du das haben willst (Startbeschleunigung), dann optimiere die Startfiles. *Wenn Du Deine Hardware generell beschleunigen willst, nimm Deinen PC in die ausgestreckten Hände, gehe zum geöffneten Fenster und lass' los. Das Ding wird unglaublich schnell auf dem Weg nach unten…*

Was ich damit sagen will: Tuningmaßnahmen gehen meist zu Ungunsten der Stabilität und Lebensdauer der Komponenten einher. Auch da gilt, dass Du wissen musst was Du tust.

Das kannst Du aber nur im *root*-Modus. Dort musst Du denken.

Es gilt (Windows+Linux) zuerst denken, dann handeln. Wenn Du Dir unsicher bist, blas' das Ganze ab und exerziere es vorerst auf einem Versuchs-PC durch.

Bild 25: Passwortabfrage unter Linux (spanisch)



Nicht umsonst gibt es in jeder Firma eine Testumgebung. Oder sollte es zumindest eine geben.

Als normaler User in Linux brauchst Du in der Regel die erhöhten Rechte nie.

Updates? Werden automatisch gemacht von sämtlicher Software…

Installieren? Was willst installieren? Oder installierst Du jede Woche einen neuen Drucker?

Wenn Du wirklich nur arbeiten willst, kommst Du unter Linux sowieso nie in die Verlegenheit als Admin (*root*) arbeiten zu müssen. Die meisten User, denen ich ein Linux draufgebügelt habe, kennen nicht einmal das *root*-Passwort. Geschweige denn dass sie wissen, was *root* überhaupt ist.

Fragen wie *„…wos moch i, wen i wo ka Berechtigung hob…“*, kontere ich immer recht gelassen mit *„…ja, dann bist dort eh falsch…“*.

Oder kennst Du in der Firma auch das Admin-Passwort von Deiner XP-Box? Eben. Geht doch.

Verzeichnisse unter Linux

Zur Linux-Rechteverwaltung gehört automatisch die Zuweisung eines geschützten Bereiches für den jeweiligen User. Dieser Bereich heißt „home Verzeichnis“.

Es handelt sich dabei um einen Ordner, wo der entsprechende User (fast) alles machen darf. Am restlichen System hat er überhaupt keinen Zugriff.

Wo liegt jetzt dieses home-Verzeichnis? Defaultmäßig unter `/home/username/`

Hier noch ein kleiner Crash-Kurs, wie im Gegensatz zu Windows die Dateistruktur in Linux aufgebaut ist.

Einmal muss ich ja das Thema anreissen, damit Du einmal weißt, wie das in Linux gehandhabt wird.

Danach wird Dir einiges logischer in Linux bezüglich Festplattenverwaltung erscheinen.

Während bei der grafischen Oberfläche die Unterschiede für den Benutzer nicht besonders groß sind, so gibt es beim Dateisystem bzw. der Verwaltung der Festplatten erhebliche Unterschiede.

Das Dateisystem von Windows besteht aus mehreren deutlich getrennten Komponenten, den einzelnen Geräten und Partitionen.

Üblicherweise werden diese Komponenten mit Laufwerksbuchstaben gekennzeichnet, also zum Beispiel A: für das Diskettenlaufwerk und C: für die erste Festplatte.

Ich glaube, das kennen eh die meisten…brauch ich kein Bild da reintun.

Unix verwaltet seine Dateien ganz ähnlich wie DOS. Das kommt schlicht daher, dass DOS in diesem Bereich viel bei Unix abgeschaut hat. Zur Erinnerung: Linux stammt von Unix ab.

In Linux werden aber sämtliche Geräte (Partitionen, Festplatten, CD-Laufwerke, Drucker, Sticks…) in Ordnern dargestellt. Immer.

Der Verzeichnisbaum beginnt immer mit dem Wurzelverzeichnis / und alle anderen Komponenten, wie zum Beispiel Partitionen und CD-Laufwerke werden in diesen Verzeichnisbaum integriert. Mittels Ordnern.

Das heißt nichts anderes, dass sämtliches Zeug in Linux (Drucker, Festplatten, Urlaubsfotos, Sexvideos, Dokumente, noch ältere Sexvideos, Treiber, Bibliotheken, Programme, USB-Sticks…) sich in einem Ordner befinden. Der Ordner heißt immer „`root`" und hat immer das Zeichen: „/". Stell Dir einen Tannenbaum vor. An der Spitze ist „/".

Nach unten hin wird der Tannenbaum immer dicker und breiter. Das sind die Unterverzeichnisse und ganz an den äußeren Spitzen die Dateien.

Zum Verständnis: *root* heißt auch der Superuser in Linux. Der Superuser ist gleichzusetzen mit dem Administrator in Windows.

„`root`" ist aber auch immer das Wurzelverzeichnis (von dem aus alles beginnt in der Dateistruktur). Also immer der Ordner ganz oben.

Root= Wurzel *…is englisch…i was*

Um die Verwirrung komplett zu machen, gibt es aber auch immer einen Ordner der „`root`" heißt, im Linux Dateibaum. Das ist das „home-Verzeichnis“ von *root*, dem Administrator. Irgendwo muss der ja auch im System sein *„Zu Hause“* haben.

/ = Wurzelverzeichnis.auch „`root`" genannt.

`/root/` = Bezeichnung des Ordners, wo der Superuser *„root“* beheimatet ist.

Root = Superuser (Administrator im Windows-Jargon)

Kennst di aus? Keine Angst, meine Mutter auch nicht…

Schau Dir das Bild an…net so noh, herst…**Bild 26**

Auf der linken Seite (die Baumstruktur) siehst Du nur Ordner. Das ist jetzt wichtig für das Verständnis. Manchen Ordnern hab ich nur andere Icons verpasst, damit ich sie visuell *(oiso…damit is schnöla ausanondakenn)* schneller zuordnen kann.

Oben siehst Du das Stammverzeichnis
(`directorio raiz` = Wurzelverzeichnis
… is spanisch… i was)

Rechts davon ist der gesamte Inhalt von diesem Wurzelverzeichnis abgebildet. Eben alles. Sieh oben in die Adressleiste. Dort steht ein Slash… heißt, ich bin ganz oben im System.

Wenn Du jetzt in der Baumstruktur den Ordner „`fat_32`" ansiehst, das ist eine FAT 32-Partition, die ich ins System eingebunden habe, damit ich darauf von Windows und Linux aus zugreifen kann.

Diese Partition wird im System als Ordner dargestellt. Dieser Ordner wird bei jedem Systemstart automatisch ins System eingebunden. Natürlich andere Partitionen und Geräte auch. Man nennt das im Fachjargon *„mounten“*.

Weiter unten siehst Du meinen Benutzernamen „`wachbirn`", der ein Unterordner von „`home`" ist. „`home`" ist bei mir auch eine eigene Partition.

Das siehst Du aber so auf die Schnelle nicht im Dateibaum. Brauchst Du eigentlich auch nicht. Im Ordner „`wachbirn`" sind alle meine Sachen drin. Man sagt, das ist mein „Home -Verzeichnis“.

Darunter gibt’s im Baum den Ordner „`BIE786G`". Das ist mein angesteckter USB_Stick. Alle Geräte und Wechselmedien die Du ansteckst, scheinen unter „`media`" auf. Alle. Du siehst auch dort den Ordner „`cdrom0`". Das ist das erste CD-Laufwerk.

Weiter unten gibt’s das „home-verzeichnis“ von root. Dort habe ich als User überhaupt keinen Zutritt. Hab ich mit rot gekennzeichnet.

Im Ordner „`tmp`" kann ich nur meine Sachen löschen. Tmp= temporärer Inhalt. Normalerweise hast Du dort eh nichts verloren.

Die zwei letzten Ordner (`Windows_7` und `Windows_xp`) habe ich nur eingebunden, damit ich von Linux aus Zugriff auf die zwei Windows-Betriebssysteme bekomme. Der PC hat nämlich Tripleboot (XP, Win7 und Debian-Linux).

In Linux sieht bis auf die 3 Ordner (`Winxp`, `Win7` und `Fat32`) defaultmäßig jeder Dateibaum so aus.

Was sollst Du Dir merken: Geräte, Festplatten und Partitionen werden in Linux per Ordner dargestellt. Immer.

Standardmäßig sind die *„eigenen Dateien“* immer unter `/home/benutzername` drin.

Das ist der einzige geschützte Bereich für den jeweiligen User, wo er seine Sachen ablegt. Überall anders hat er sowieso keine Rechte, irgendwas zu deponieren.

Wie das vom System mit den Ordnern und Partitionen gehandhabt wird… behandel ich ein anderes Mal. Technisch heißt der Ausdruck dafür *„mounten“*. Aber ich will darauf jetzt nicht näher eingehen, sonst wird’s zu viel für den Anfang.

Ordner mit grünem Hakerl geben mir Vollzugriff. Die restlichen Ordner sind systemrelevant. Dort habe ich maximal Leserechte. Als `root` natürlich alle Rechte. Als User brauch ich dort sowieso nix. Also egal.

Wenn Du als `root` auf dem PC Deines Freundes das eingibst „`rm -rf /*`" *(read mail really fast…:-)*…dann hast Du es geschafft. Wahrscheinlich wird er dann mit geschwollenen Halsschlagadern und wutverzerrtem Gesicht versuchen, Deines durch mechanische Einwirkung seiner Tastatureingabewerkzeuge upzudaten.

`rm -rf` löscht nämlich das komplette Linux-System. Wenn Du in einem Netzwerk auch noch hängst, versucht es natürlich auch, die entfernten Knoten zu löschen. Hängt ja alles unter „/".

`rm` = *remove* = löschen

`-rf` = *recursive, force* = ohne Nachfrage…

`/*` = alles unter dem Stammverzeichnis /

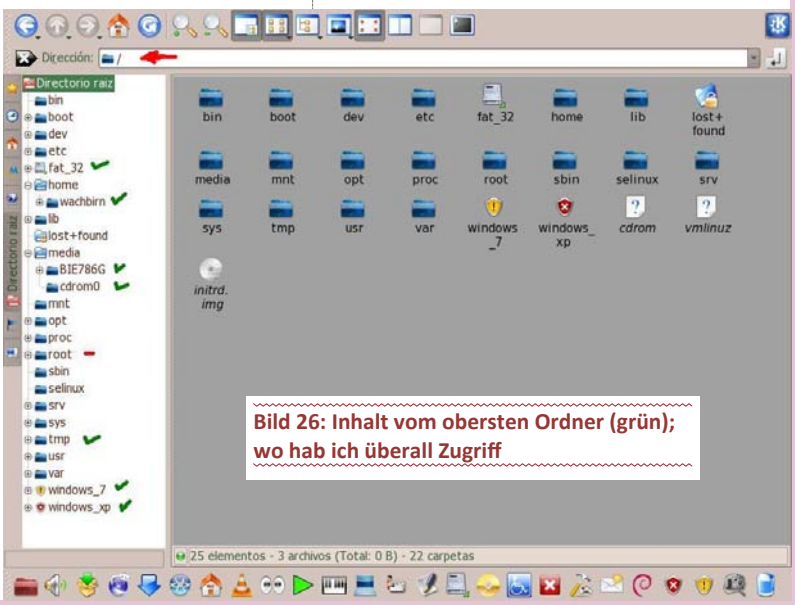


Bild 26: Inhalt vom obersten Ordner (grün); wo hab ich überall Zugriff

Und nein, da kommt vorher keine Abfrage, ob Du wirklich löschen willst. Du bist root. Du musst wissen, was Du tust.

Erinner' Dich an die Anekdote vom Linux-Kurs… das war der Befehl.

Wenn Du Linux zerstören willst, kein Problem. Du kannst machen, was Du willst. Aber gib anderen nicht die Schuld für Deine Aktionen.

Goldene Regel: Zuerst denken, dann tippen.

Was Du Dir nur merken brauchst: *…scho wida wos…*

In Linux geht alles vom Wurzelverzeichnis aus weg. Man sagt auch, es geht von „root" aus weg. Damit ist immer das oberste Verzeichnis „/" gemeint.

In Windows hast Du üblicherweise mehrere Dateibäume. Kommt auf die Partitionen und Geräte an. Den obersten Punkt (die Spitze des Tannenbaumes) bildet immer ein Laufwerksbuchstabe.

Deshalb gibt es immer mehrere „Tannenbäume" (Dateibäume) in einem Windows-System.

Wenn Du Dich im System immer eine Ebene hinaufklickst, kommst Du bei Linux letztendlich immer zum Startordner „/", (egal, ob Du Dich gerade auf einem USB-Stick, dem CD-Laufwerk oder der siebenten logischen Partition befindest) bei Windows zu einem von vielen Laufwerksbuchstaben.

Da Windows nicht so streng hierarchisch aufgebaut ist, „muss" es seine obersten Einstiegspunkte (die Spitze des Tannenbaumes) eindeutig kennzeichnen. Windows macht das mit Buchstaben.

Linux braucht den obersten Einstiegspunkt nicht mit einem bestimmten Buchstaben kennzeichnen, da dieser immer derselbe ist. Nämlich root „/".

Linux ist im Gegensatz zu Windows streng hierarchisch aufgebaut.

Linux = ein Tannenbaum (Dateibaum)

Windows = mehrere Tannenbäume (Dateibäume, ergo mehrere Buchstaben)

Nur die Registry in Windows ist hierarchisch aufgebaut.

Der Ordner „root" in Linux gehört nur dem Superuser *root*… Das ist das Home-Verzeichnis von *root*.

Die Rechte des besonderen Benutzers *root* holt man sich immer nur dann, wenn man sie wirklich benötigt, also zum Beispiel für die Installation von Programmen. Dadurch kann ein Benutzer unter Linux nicht so leicht das System verändern oder Schadprogramme installieren.

Bei Windows benötigen viele Programme Administratoren-Rechte, weil sie in irgendeiner Form in die Registry schreiben wollen.

Die Registry ist eine zentrale Stelle für Konfigurationseinstellungen. Hier finden sich sowohl die globalen Einstellungen, als auch die individuellen Einstellungen der Benutzer.

Die Registry wird im Laufe der Zeit immer größer, da nahezu jede Software-Deinstallation (meist) ihre Spuren dort hinterlässt.

Es ist zwar kein erquickliches Gefühl, wenn man weiß, dass im Hintergrund die Registry sich täglich immer mehr aufbläht. Aber mittlerweile haben wir gelernt, damit zu leben.

Die Tuningtools jeglicher Art verschaffen zumindest am Anfang etwas Luft. Aber die Risiken sind mir zu hoch. Sämtliche Tools greifen naturgemäß sehr tief ins System ein. Fehlerquellen wie Fehlfunktionen, gelöschte Dateipfade bis zu abgebrochenen Sicherungen und was weiß ich noch… empfehlen tu ich keines. Wennst die Registry aufräumen willst, mach's manuell. Im Gegensatz zum Tool weißt Du da genau, was Du machst. Hoffentlich…

Rechteverwaltung unter Linux

Weil Linux ein Mehrbenutzersystem ist, ist es natürlich notwendig, dass jeder User sich beim System mit einem Namen anmeldet und mit einem Passwort legitimiert. Das Betriebssystem ist sich also immer im Klaren, wer gerade wo arbeitet.

Warum reite ich da drauf immer so herum?

Bei Linux gibt es keine Home und Professionell-Unterscheidungen wie bei Windows. Egal, welches Linux Du benutzt, Du arbeitest immer mit der vollen (selben) Rechteverwaltung. Auch wenn Du nur „alleine" am System arbeitest.

Du kannst logischerweise auch mit jedem Linux einen Server für 5000 Clients installieren oder ein Einzelplatzsystem. Die Rechteverwaltung ist immer dieselbe.

Bei Windows gibt es dagegen erhebliche Unterschiede zwischen der Home und Enterprise-Version. Vor allem in der User- und Rechteverwaltung.

Früher haben wirklich mehrere Personen auf einem System gleichzeitig gearbeitet. Naheliegend, dass man da als normaler User zum Beispiel nicht so einfach die Kiste ausschalten durfte. Hätten die anderen keine Freude gehabt. Das lässt sich eben nur mit strikter Benutzerverwaltung realisieren.

Windows realisiert diese fehlende Funktionalität (Mehrbenutzersystem) mit Windows-Terminalservern. Natürlich gegen Aufpreis…

User werden zwar mit ihrem Namen verwaltet, intern arbeitet Linux aber mit Usernummern, genauso wie Windows. Jeder User hat also eine UserID, kurz oft UID genannt. Sein Name ist nur Beiwerk, das jederzeit verändert werden kann. Jeder User ist Mitglied mindestens einer Gruppe. Diese Gruppe ist immer die Hauptgruppe.

Merke: Unter Windows hat alles einen Besitzer.

In Linux gehört jede neu angelegte Datei dem jeweiligen User und ist auch gleichzeitig Mitglied der Hauptgruppe.

Es kann beliebig viele Gruppen in einem System geben, auch sie haben intern Nummern (GroupID oder GID). Im Prinzip sind Gruppen nur eine Möglichkeit, noch detailliertere Einstellungsmöglichkeiten zu haben, wer was darf.

Das System von Usernamen und Passwort ist auch von Windows her bekannt, aber bei Linux spielt es eine völlig andere Rolle. Jede einzelne Datei im System hat einen Eigentümer (einen User, gewöhnlich der, der die Datei angelegt hat) und gehört zu einer Gruppe.

Jede Datei hat aber auch eine Einstellung, wer sie lesen, überschreiben (verändern) oder ausführen darf. (Richtet sich nach Eigentümer, Gruppenzugehörigkeit und „Rest der Welt") Es ist also möglich, genauestens festzulegen, wer welche Dateien bearbeiten darf, wer sie lesen darf und wer nicht.

Ein System, das solche Einschränkungen erlaubt braucht zwingend jemand, der diese Einstellungen erledigt und daher außerhalb ihrer Wirkung steht. Kurz gesagt einen Superuser, der Systemverwalter. Bei Linux heißt der Superuser gewöhnlich root, er hat immer die UserID (UID) 0.

Deshalb ist bei Linux der grafische root-Login meist unterbunden. Könnte man zu viel kaputt machen…

Im Gegensatz zur Registry in Windows ist das bei Linux generell anders gelöst. Programme legen ihre Konfigurationseinstellungen grundsätzlich in menschenlesbaren Textdateien ab. Diese Dateien liegen im Verzeichnis /etc und heißen meistens so wie die Anwendung die mit ihnen konfiguriert wird.

Benötigt eine Anwendung mehrere Konfigurationsdateien, dann erstellt sie ein Verzeichnis unterhalb von /etc.

etc = *editable text configuration* = veränderbare Konfigurationsdateien

Verschiedene Programme und sogar verschiedene Programmversionen können damit Konfigurationseinstellungen ablegen, ohne sich gegenseitig zu stören.

Das heißt, man kann problemlos eine ältere Version eines Programmes neben einem neuen gleichzeitig am System laufen haben.

Die persönlichen Konfigurationseinstellungen der Benutzer landen natürlich nicht im Verzeichnis /etc/, die Benutzer haben dort keine Schreibrechte.

Die persönlichen Konfigurationseinstellungen landen immer im persönlichen Verzeichnis, dem Homeverzeichnis. Auch dort heißen die Dateien so ähnlich wie die Anwendung, nur dass ihnen hier ein Punkt an den Anfang des Dateinamens gestellt wird.

Der Punkt am Anfang des Datei- oder Verzeichnisnamens kennzeichnet eine verborgene Datei, die in den normalen Auflistungen nicht mit angezeigt wird. Dadurch sind die Konfigurationsdateien nicht sichtbar, wenn man sich ganz normal den Inhalt des Homeverzeichnisses anschaut. Wenn man auch diese Dateien sehen will, dann muss man den entsprechenden Schalter im Dateimanager aktivieren.

Wenn ein Benutzer jetzt ein Anwendungsprogramm startet, dann sucht das zuerst im Homeverzeichnis des Benutzers nach seiner Konfiguration und wenn es die nicht findet dann auch im Verzeichnis /etc, ein leicht nachvollziehbares Verfahren. Auch wenn man einmal die Konfiguration eines Programmes vollständig verbogen hat ist eine Reparatur kein Problem. Man löscht einfach die Konfigurationsdatei oder den Konfigurationsordner im Homeverzeichnis und schon startet das Programm wieder mit den Standardeinstellungen.

Vorteilhaft finde ich unter Linux noch, dass jede Festplatte (Partition) automatisch 5 % seines Speicherplatzes nicht belegt. Der Platz ist für den Superuser *root* gedacht, damit der bei einer eventuell vollen Festplatte sich immer noch anmelden und das System aufräumen kann.

Wie sieht das jetzt mit den Rechten aus? So…

```
… # ls -l

drwxr-xr-x 2 wachbirn user 4096 2008-11-07
17:00 XYZ

-rwxr-xr-- 1 wachbirn user 256 2008-11-07
17:00 ABC.tar
```

Wir sehen hier eine typische Ausgabe des Shellbefehls ls

Natürlich kann man sich die Rechte auch grafisch anzeigen lassen. Aber lass Dir gesagt sein, dass es auf der Konsole sehr übersichtlich ist und vor allem sämtliche Aktionen sehr schnell durchgeführt werden.

Sprich > Konsole ist schnell und übersichtlich, speziell bei der Rechteverwaltung.

Bei der oberen Zeile handelt es sich um ein Verzeichnis. Das wird durch das d (directory)

ganz am Anfang der Zeile verdeutlicht. Ein weiterer Hinweis ist die fehlende Dateiendung.

Die Ausgabe von `ls` mit dem Schalter `l` (*long* = lange Ausgabe) erzeugt, wie wir sehen, eine Ausgabe in mehreren Spalten.

1. Spalte: Hier steht die Auflistung der Rechte, die für das Verzeichnis (Rechtecode beginnt mit einem d) oder die Datei (Rechtecode startet mit -) gesetzt wurde.

2. Spalte: Hier findet sich die Zahl der Unterverzeichnisse. (Handelt es sich um eine Datei, steht hier immer eine 1, im Falle eines Verzeichnisses immer mind. eine 2, da in einem Verzeichnis immer das übergeordnete und das aktuelle Verzeichnis gezählt werden > daher immer mindestens die Zahl 2 bei Ordnern)

3. Spalte: Hier steht der Eigentümer, dem die Datei oder das Verzeichnis gehört.

4. Spalte: Hier steht die Gruppe des Eigentümers.

5. Spalte: Hier steht, wie unschwer zu erkennen ist, das Datum und die Uhrzeit der letzten Änderung der Datei oder des Verzeichnisses.

6. Spalte: Hier findet sich schlussendlich der Datei- oder Verzeichnisname.

Wie setzen sich die Rechte zusammen: **Bild 27**

Für den folgenden Abschnitt betrachten wir uns zunächst einmal die erste Spalte des Beispiels von oben.

```
d rwx r-x r-x
```

Im Falle des Verzeichnisses XYZ wurden die Rechte wie folgt gesetzt:

`d` Kennzeichnung für das Verzeichnis. > d = *directory* = englisch für Ordner

`rwx` Die Rechte für den Eigentümer der Datei. Im Fall des Beispiels ist das der Benutzer `wachbirn`.

`r-x` Die Rechte für die Gruppe, in der sich der Eigentümer (alias `wachbirn`) befindet.

`r-x` Die Rechte für den Rest der Welt oder besser für alle anderen Benutzer auf diesem LinuxSystem.

Doch was bedeuten nun `rwx`:

`r` steht für *read* (engl. für Lesen) und bedeutet somit: Ist dieses Recht gesetzt, darf gelesen werden!

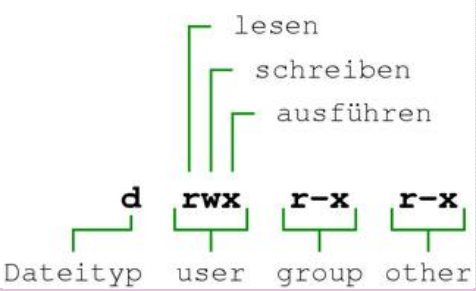
`w` steht für *write* (engl. für Schreiben) und sagt somit aus: Hier darf verändert (geschrieben) werden!

`x` steht für *execute* (engl. für Ausführen): Es darf ausgeführt werden!

Für Verzeichnisse bedeutet dies: Ist das `x` gesetzt, darf in dieses Verzeichnis gewechselt werden. Ist das `x` nicht gesetzt, ist dies nicht möglich.

Für Dateien bedeutet dieses `x`, dass ausführbare Dateien (ELF-Dateien, das Linux-Pendant zu Exe oder auch ausführbare Bash-Scripte) ausgeführt werden können. Dateien, für die das `x` nicht gesetzt ist, sind also für den entsprechenden Benutzer nicht ausführbar.

Bild 27: Schematische Rechtestruktu in Linux



Das merkst Du in Linux auch gut daran, dass sämtliche vom Internet heruntergeladene Linux-Programme einmal nicht ausführbar sind. Weil das Ausführen-bit fehlt.

Ist so was wie eine Sicherheitseinrichtung. Erst wenn das `x` dabei ist, kannst Du wie bei einer Windows-exe ausführen.

Wie bekommst Du das Ausführungsbit?

So: Befehl:

```
> chmod +x /pfad/zu/ausführbarer Datei
```

`chmod` = *change mode*

`+x` = hänge das Ausführungsbit dazu

Jetzt die Rechteverwaltung aus Betriebssystemsicht

Die Rechte in Bitform

Nachdem wir nun wissen, wie sich die Rechte auswirken und wie sie vom Betriebssystem ausgegeben werden, schauen wir uns nun an, wie das Betriebssystem selber die Rechte versteht, bzw. wie diese abgespeichert werden.

Wenn wir uns die Ausgabe aus dem Beispiel noch einmal genau anschauen, könnte einem eines auffallen.

Rechte rwx r-x

Binaer 111 101

Oktal (4+2+1=)7 (4+1=)5

Wenn wir uns jetzt das Binäre Zahlensystem, auf dem ja die Informatik aufbaut, ins Gedächtnis rufen, fällt uns eines auf: Jede der 3 Rechtegruppen lässt sich eine Zahl zuordnen. In unserem Beispiel entspricht die Rechteverteilung also dem Schlüssel 755.

Zur Erinnerung: d|rwx|r-x|r-x =755

7 + 5+ 5 = 755

Nachdem wir nun die drei Formen kennen, in denen die verschiedenen Rechte auftreten können, gehen wir einen Schritt weiter.

Mit welchen Rechten werden neue Verzeichnisse und Dateien erzeugt:

Um das zu klären, müssen wir uns zwei Dateien anschauen, in denen sich alle Einstellungen befinden, die für die Shell von Bedeutung sind. Manchmal kann es aber auch sein, dass diese Info in der Datei „`profile`" steht.

`/etc/bashrc` (die globale Konfigurationsdatei)

`~/.bashrc` (die lokale Konfigurationsdatei im Homeverzeichnis des Benutzers)

Dort findet sich ein Eintrag in folgender Form: `umask 022`

Siehe **Bild 28**

Das mag auf den ersten Blick verwunderlich wirken, aber dieser Eintrag ist dafür zuständig, dass neu angelegte Dateien und Verzeichnisse die entsprechenden Rechte bekommen.

Die Form, in der ein Verzeichnis oder eine Datei maximal mögliche Rechte hat (lesen/ schreiben/ ausführen für den Eigentümer/ dessen Gruppe/ alle anderen), ist die 777, da die 7 den Rechten `rwx` (vgl. Rechte in Bitform) entspricht.

Die grundlegende Funktion der `umask` ist es nun, von der maximal möglichen Rechteausstattung abgezogen zu werden.

Hier gibt es aber eine Unterscheidung zwischen Dateien und Verzeichnissen.

Verzeichnisse benötigen Ausführungsrechte (x mit der Wertigkeit 1 (siehe Position im Bitmuster)). Folglich muss die `umask` im Fall eines Verzeichnisses von 777 abgezogen werden. Die normale Voreinstellung für die `umask` ist 022. Damit ergibt sich als Standardrechte für neu angelegte Verzeichnisse die 755 (was gleichzeitig den Rechten im Beispiel entspricht).

Dateien benötigen im Normalfall keine Ausführungsrechte. Daher werden in diesem Falle nur Lese- und Schreibrechte gewertet (4+2=6) und die Standard-`umask` 022 von den maximal nötigen Dateirechten 666 abgezogen. Damit hat eine neu erzeugte Datei standardmäßig die Rechte 644. Sprich, der Ersteller darf alles mit der Datei, alle anderen nur lesen.

Wie ändere ich jetzt die Rechte?

chmod

Die Rechte an einer Datei ändert man mit dem Programm `chmod`. *(change mode)*

Wer schon einmal mit FTP-Programmen hantiert hat, kennt das sicher schon.

Es gibt verschiedene Möglichkeiten, `chmod` mitzuteilen, was geändert werden soll.

Die einfachste Version ist folgende:

```
chmod 777 /home/wachbirn/ordner
```

Dieser Befehl setzt die Rechte am Verzeichnis „`/home/wachbirn/ordner`" auf `drwxrwxrwx`.

Den selben Befehl hätte ich auch anders schreiben können:

```
chmod u +a,g +a,o +a /home/wachbirn/ordner
```

In diesem Fall werden dem User `u`, wie auch der Group `g`, wie auch allen anderen `o` sämtliche Rechte gegeben.

```
+a = alle Rechte
```

`chmod` kann sowohl mit den Endwerten in dezimaler Schreibweise die Rechte annehmen oder die einzelnen Rechte in Flagform verwenden.

Möchte ich ausgehend von 770 mit Hilfe von `chmod` den anderen Benutzern der Gruppe des Eigentümers sämtliche Rechte entziehen, geht das auf folgenden Wegen auch:

```
chmod 700 /home/wachbirn/ordner
```

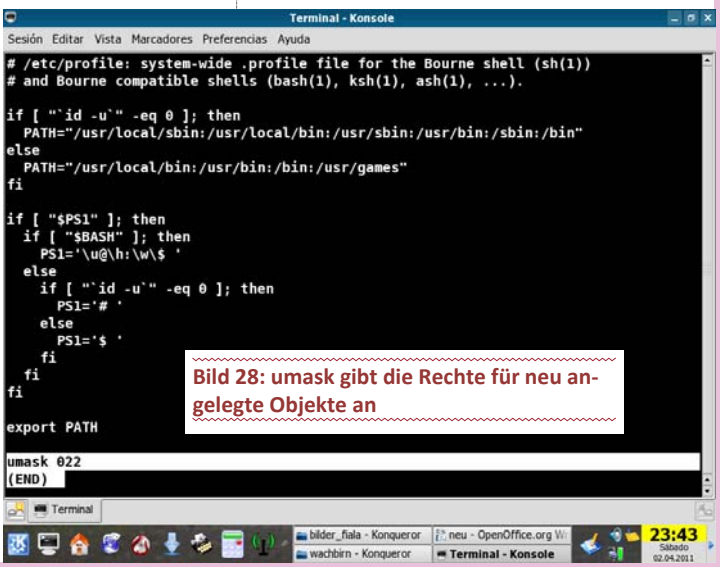


Bild 28: umask gibt die Rechte für neu angelegte Objekte an

```
chmod g -a /home/wachbirn/ordner

chmod g -r,g -w,g -x /home/wachbirn/ordner
```

Das Endergebnis ist immer dasselbe. Nur der Ersteller hat alle Rechte (auf den Ordner „ordner"), die anderen überhaupt keine.

Man sieht, ich kann demnach einzelne Rechte entweder durch + hinzufügen oder unter - wieder wegnehmen.

Aufpassen: Wenn Du versehentlich einem der wichtigen Verzeichnisse /etc, /root, /sbin und noch ein paar anderen die Ausführrechte entzieht (also das Betreten des Ordners unterbindest), wirst Du danach ein paar Stresswimmerln bekommen. Zusätzlich kannst Du auch gleich neu installieren.

Du bist *root*, Du musst wissen was Du tust.

Speziell der Rechteentzug im Ordner /sbin (dort sind viele wichtige Admin-Tools drin, unter anderem chmod :-), ziehen einen sicheren Exitus des Betriebssystems nach sich.

chown

Wie wir weiter oben schon gesehen haben, ist es nicht ganz unerheblich, welcher Benutzer Eigentümer eines Verzeichnisses oder einer Datei ist, weil sich auf diesen Umstand die vergebenen Rechte sehr direkt beziehen. Aus diesem Grunde sollte man sich Gedanken machen, wer der Eigentümer von Dateien und Verzeichnissen ist und wer sich unter Umständen mit ihm noch in ein und derselben Gruppe befindet.

Diesen Gedankengang solltest Du natürlich auch in Windows beherzigen.

Im Normalfall werden Dateien und Verzeichnisse erstellt und das Betriebssystem weist der Datei oder dem Verzeichnis als Eigentümer den Benutzer zu, der diese Datei oder das Verzeichnis erstellt hat.

Daraus ergibt sich auch die Gruppenzugehörigkeit, da automatisch die primäre Gruppe eines Benutzers auch als Gruppe für die Datei oder das Verzeichnis genutzt wird.

Mit chown kann der Eigentümer und/oder die Gruppe einer Datei oder eines Verzeichnisses geändert werden. Das Recht, diese Operation auszuführen, haben immer der bisherige Eigentümer und natürlicherweise der *root*-Account. *(i bin root, i derf des…)*

Die Syntax von chown sieht wie folgt aus:

```
chown neuer_Benutzer:neue_Gruppe /pfad/zur/Datei
```

Möglich sind hierbei: neuer_Benutzer wenn nur der Eigentümer geändert werden soll, neuer_Benutzer:neue_Gruppe wenn beide geändert werden sollen und :neue_Gruppe wenn nur die Gruppe geändert werden soll.

Damit sind alle grundlegenden Optionen für das Rechtesystem erklärt.

Viel mehr wirst Du auf der Linux-Seite als User, wenn überhaupt, nicht brauchen.

Es sei noch einmal darauf hingewiesen, dass Du das ganze Zeugs auch grafisch machen kannst. Mach es trotzdem immer unter der Kommandozeile. Wenn Du für eine 2 Terrabyte-Platte den Besitz übernehmen willst, dauert das auf der Kommandozeile ein paar Sekunden. Bei der grafischen Lösung immer bedeutend länger. (bis der Befehl abgearbeitet ist…)

Hier noch zwei Screenshots von der grafischen Lösung. **Bild 29**

Es handelt sich dabei um einen Ordner mit dem Namen „allacher". Ganz oben steht der Besitzer…also ich. Darunter was die Gruppe und der Rest der Welt machen dürfen. Alle außer dem Eigentümer dürfen den Ordner nur einsehen. Nicht mehr.

Die umask 0022 kommt hier zum Tragen. Heißt für Verzeichnisse demnach > 755

Im Ordner allacher ist eine pdf-Datei. Dort gelten natürlich dieselben Rechte. Da eine Datei normalerweise nicht ausführbar sein sollte *(wo wüstn a higeh mit ana pdf-Datei…muasst di nur genieren wahrscheinlich :-))*, kommen dort die Rechte 644 zum Tragen. **Bild 30**

Noch einmal zum Mitdenken:

die vollen Rechte hab ich immer mit dieser Schreibform > 777

der erste 7er steht für den Besitzer, der 2te für die Gruppe, der Dritte für den Rest der Welt. *(san eh nimma so füh üba…)*

lesen= 4

schreiben(ändern und löschen)=2

ausführen= 1

Meine umask lautet: 0022

Heißt für Ordner: 755

Heißt für Dateien: 644

Ordner brauchen das Ausführungsbit, da man sie sonst nicht betreten kann.

Dateien brauchen dieses Bit normal nie.

Eselsbrücke: Bei Ordnern hast Du immer ungerade, bei Dateien immer gerade Zahlen (für eine vernünftige Arbeitsweise)

Das wars…in der grafischen Aufbereitung ist das meist „ausgedeutscht". Siehe die 2 Screenshots.

Bild 29: Zugriffsrechte in grafischer Form

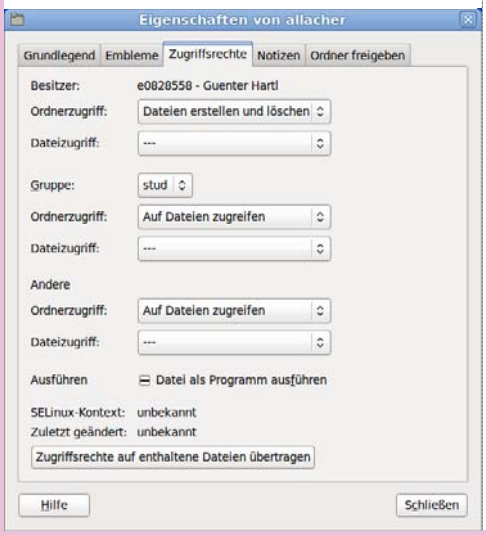
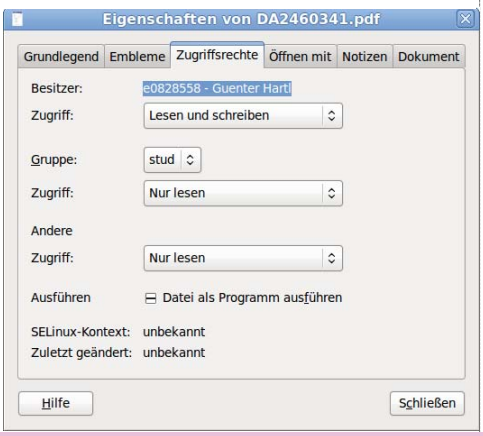


Bild 30: Ich darf lesen und schreiben, alle anderen nur lesen; ich bin der Besitzer



Bei pdf dürfen alle außer dem Besitzer nur lesen (Oktalwert 4). Besitzer darf lesen und schreiben (Oktalwert 4+2=6).

Bei den Ordnern darf der Besitzer alles (lesen+schreiben+ausführen=7), alle anderen nur betreten und lesen (betreten = x=1…lesen =4…ergibt 5)

Der Vollständigkeit halber sei hier noch erwähnt, dass es auf der Linux-Seite ein Tool gibt, mit dem ein User automatisch mit höheren Rechten arbeiten kann.

Ein Admin kann zum Beispiel einem User bestimmte Aufgaben übertragen(User anlegen…), ohne jetzt das root-Passwort herausgeben zu müssen.

Das Tool heißt „sudo".

Da die Anwendung des oben erwähnten Tools einiges Wissen in der Rechteverwaltung und besonders in der Userverwaltung von Linux voraussetzt, lass ich das jetzt mal so im Raum stehen. Sonst brauch ich eine eigene Ausgabe der PCNEWS nur für diesen Artikel…da Fiala hat eh schon so genug Arbeit…

Ist alles sehr verzahnt, aber irgendwo muss ich mal abgrenzen, sonst komm ich da nie weiter…

Linux arbeitet genauso wie Windows mit internen Bezeichnungen. Als User hast Du immer eine eindeutige User-ID

Ich sitze gerade an der Uni auf einem Linux-Rechner…sieh Dir die Zeilen an:

```
e0828558@pc010:~$ id

uid=61789(e0828558) gid=35000(stud)
Gruppen=35000(stud)

Befehl id > Identität

uid > User id

gid > meine Hauptgruppe
```

Alle Files, die ich anlege, bekommen immer diese uid und gid.

Im privaten Umfeld fangen die uid meist bei 1000 an. Die uids von 0-999 sind immer dem System zugeordnet. Ab 1000 fangen die User an.

Also der nächste angelegte User hätte die interne User-Id 1001…usw.

Externe Datenträger

Der Zugriff auf eine externe NTFS-Platte unter Linux fällt nicht ganz so robust aus wie jener unter Windows. Wenn das Laufwerk nicht ordentlich ausgeworfen wurde, weigert sich Linux mit hoher Wahrscheinlichkeit, mit der Platte zu kooperieren. Das passiert meist, wenn die externe Platte nicht ordnungsgemäß vom Windows-System entfernt wurde.

Da Linux keine Möglichkeit der Dateisystemüberprüfung von NTFS hat *(closed source)*, wird so was mit einer Fehlermeldung quittiert.

Tritt dieser Fall ein, helfen manchmal die Linux-Tools (ntfsfix…). Falls nicht, einfach ein Windows starten, Platte anhängen und ordnungsgemäß wieder aushängen.

Dann kann man die Platte in der Regel wieder überall verwenden. Üblicherweise merkt man unter Windows nichts davon, dass die Platte nicht einwandfrei vom System entfernt wurde.

Darum predige ich auch immer das ordnungsgemäße Entfernen von externen Geräten… irgendwann kracht's halt, wenn man sich nicht dran hält. (und verliert im schlimmsten Fall seine Daten).

Linux ist ein Unix-Klon

Unix besitzt ein zweistufiges Benutzermodell. Es gibt den Benutzer *root*, der alles darf, und alle anderen Benutzer, deren Rechte erheblich eingeschränkt sind. Muss ein Benutzer Aktionen ausführen, die erweiterte Rechte erfordern, kann er dies nur tun, wenn er die Rechte des Benutzers *root* bekommt. Das ist auch heute noch so und führt zu erheblichen Schwierigkeiten bei der Implementierung von Sicherheitskonzepten.

Windows NT bekam eine andere Sicherheitsarchitektur als Unix. Auch Windows besitzt einen allmächtigen Benutzer mit dem Namen SYSTEM.

Es ist allerdings nicht möglich, sich unter diesem Namen anzumelden. Alle anderen Benutzer besitzen verschiedene globale Rechte. Die meisten Rechte können Benutzer in Anspruch nehmen, die der Gruppe Administratoren angehören. Sie haben fast so viele Rechte wie der System-Benutzer. Das bekannteste (und wahrscheinlich Wichtigste) Recht ist *„Übernehmen des Besitzes von Dateien und Objekten"*.

Ein Administrator, der versucht, auf eine Datei zuzugreifen, für die er keine Berechtigung besitzt, bekommt, wie jeder andere Benutzer auch, eine Fehlermeldung angezeigt.

Allerdings besitzt er das Recht, die Datei zu seinem Eigentum zu machen. Dies geht bequem über den Windows-Explorer. Anschließend kann er sich selbst Berechtigungen zuweisen und die Datei nach Belieben nutzen.

Ist immer dasselbe Prinzip: Zuerst Besitz übernehmen, dann die Rechte vergeben.

Grundsätzlich hat Microsoft bei der Entwicklung von Windows NT erkannt, dass das Superuser-Modell von Unix für moderne verteilte Rechnersysteme zu einfach gehalten ist. Unter Zeitdruck im Wettlauf mit IBM dachte Microsoft viele Gedanken nicht zu Ende und machte Kompromisse und Fehler, die bis in die aktuellen Windows-Versionen nachwirken.

Ganz anders ist das unter Unix. In den 80er Jahren konnten sich Unix-Anwender auf einem Großrechner nicht als *root* anmelden. Erst als Linux Mitte der 90er Jahre größere Popularität erlangte, verbreitete sich Unix auf Personal-Computern, bei denen der Benutzer gleichzeitig Computerbesitzer ist und das Root-Passwort kennt.

Unix-Entwickler schreiben Anwendungsprogramme in der Regel so, dass Root-Rechte weder für die Installation noch für die Ausführung erforderlich sind. Das Konzept ist einfach: Die Anwendungen können in allgemein zugänglichen Verzeichnissen, beispielweise /bin oder /usr/bin, installiert werden. Sie lassen sich jedoch auch in privaten Verzeichnissen einrichten, etwa /home/user/bin. Meist gibt es zwei Konfigurationsdateien, eine im Verzeichnis /etc, die der Superuser verwaltet, eine weitere im Home-Verzeichnis des Benutzers. Ist ein Parameter in beiden Dateien angegeben, so hat die des Benutzers Priorität.

Windows Registry

Unter Windows ist eine ähnliche Vorgehensweise mit der Registry realisierbar. Sowohl MS-DOS-basierte Windows-Versionen als auch Windows-NT-basierte Versionen haben die beiden Registry-Äste HKEY_LOCAL_MACHINE und HKEY_CURRENT_USER.

Desktop-Anwendungen könnten versuchen, Default-Werte in HKEY_LOCAL_MACHINE zu schreiben. Benutzereinstellungen kommen hingegen in HKEY_CURRENT_USER. Scheitert das Schreiben in HKEY_LOCAL_MACHINE mangels Berechtigungen, dann gibt es entweder keine Default-Werte oder die Anwendung fordert die erhöhten Rechte ein (mittels UAC).

Dieses einfache Prinzip verhindern allerdings die Microsoft-Logo-Programme. Um das begehrte Logo zu erhalten, müssen Programme unter anderem Uninstall-Information in HKEY_LOCAL_MACHINE schreiben. Das bedeutet im Klartext: Ohne Administratorrechte gibt es keine Logo-konforme Installation.

Die Philosophien der 2 Betriebssysteme…

Unter Linux heißt der Administrator *root* und hat die interne User-ID 0. Er wird durch keinerlei Zugriffsrechte eingeschränkt.

Unter Windows heißt der Administrator *„Administrator"*. Er kann noch durch Zugriffsrechte eingeschränkt sein.

Unter Linux gilt: An der grafischen Oberfläche grundsätzlich nicht als Administrator anmelden, sondern bei Bedarf mit „su" oder ähnlichen Mechanismen nur für ein Programm oder Fenster auf den Administrator umschalten. Es ist absolut unüblich (und auch nicht notwendig), dass normale Anwender mit Administrator-Rechten arbeiten.

Unter Windows bleibt einem gar nichts anderes übrig, als sich an der grafischen Oberfläche als Administrator anzumelden, wenn man den Rechner verwalten will.

Die Anmeldung auf den Windows-Versionen Win95/98/98SE/ME war gar keine richtige Anmeldung, weil die Benutzer intern nicht unterschieden wurden und zum Beispiel Dateien keinen *„Besitzer"* hatten. Aber dieses Thema sollte heutzutage sowieso nicht mehr relevant sein.

Bei Windows NT/2000/2003/2008/XP/Vista/7 wird inzwischen streng getrennt zwischen dem Administrator und den einzelnen Benutzern. Allerdings gibt es keinen Zwang, sich auf einem Rechner anzumelden, er kann zum Beispiel mit einem Standard-Benutzer ohne Passwort eingerichtet sein. Viele private Anwender machen dies auch so oder arbeiten gleich als Administrator, da sie in der Regel der einzige Benutzer des Windows-Rechners sind.

Unter Linux muss man sich stets mit einem Benutzernamen + Passwort anmelden. Somit weiß das System immer, wer man ist. Dem Anwender wurden beim Einrichten vom System-Administrator gewisse Rechte oder Verbote erteilt. Er kann auf dem System genau diejenigen Dateien einsehen, ändern oder löschen, wie es ihm vom Administrator erlaubt wurde. Die eigentlichen Systemdateien darf er beispielsweise nicht ändern. So kann man nicht aus Versehen das System beschädigen, Konfigurations-Dateien löschen oder gar die Festplatte formatieren. Von ihm neu angelegte Dateien und Verzeichnisse werden automatisch diesem Benutzer zugeordnet.

Man kann unter Linux auch nicht auf die Dateien von anderen Anwendern zugreifen oder diese einsehen (ob lokal oder über das Netz), außer es wurde explizit von diesen erlaubt. Grundsätzlich gilt folgende Regel: *„Es ist alles verboten, das nicht ausdrücklich erlaubt ist"*. Daher bitte nicht wundern, wenn ab und zu die Meldung *„Permission denied"* angezeigt wird.

Häufig wird unter Windows bereits beim Download einer Datei im Web-Browser die Extension interpretiert und das Programm im Falle einer EXE-Datei zum Beispiel sofort zur Ausführung angeboten. Dies ist zwar sehr bequem, aber auch sehr unsicher.

Bei Linux wird das Ausführungsrecht nicht mit übertragen, d. h. Downloads sind nicht sofort ausführbar, sondern müssen erst gespeichert und dann mit dem Ausführungs-Recht versehen werden.

Windows und Linux verwenden unterschiedliche Dateisystemtypen. Windows 95/98/98SE/ME arbeitet ausschließlich mit den Dateisystemen FAT12, FAT16, FAT32 und VFAT.

Windows NT/2000/2003/2008/XP/Vista/7 verwendet primär das Dateisystem NTFS (in unterschiedlichen Varianten). Unter Linux ist das Dateisystem Ext3/4 am weitesten verbreitet, aber es gibt noch viele weitere wie zum Beispiel JFS, XFS und ReiserFS, BtrFS, die alle sogenannte „protokollierende Dateisysteme" *(journaling filesystems)* sind (wie NTFS auch). Diese bieten eine höhere Fehlertoleranz und müssen nach einem Systemabsturz beim Hochfahren keinen umfangreichen Prüfdurchlauf durchführen.

Linux kann auf jeden Fall immer auf alle Windows-Dateisysteme lesend und schreibend zugreifen während umgekehrt Windows leider nicht auf alle Linux-Dateisysteme zugreifen kann.

Unter Windows gibt es pro Partition bzw. pro verbundener Netzwerk-Freigabe einen Laufwerksbuchstaben A:, B:, C:, D:,… mit jeweils einem eigenen Dateibaum. Zumindest Laufwerk C: existiert immer.

Unter Linux gibt es nur einen einzigen Dateibaum, in dem die Dateisysteme der einzelnen Partitionen in bestimmten (leeren) Verzeichnissen „eingehängt" (montiert) werden. Startpunkt ist die „root"-Partition, die beim Booten vom Kernel montiert wird. In den Dateibaum dieser Partition werden nach Bedarf gegebenenfalls weitere Partitionen eingehängt.

Bevor unter Linux ein Datenträger (Platte, CD, Diskette, USB-Stick,…) benutzt werden kann, muss er „eingehängt" (montiert) werden. Analog muss er „ausgehängt" (demontiert) werden, bevor er wieder entfernt werden darf. Das Dateisystem auf dem Datenträger erscheint an einer (meist wählbaren Stelle) des Gesamt-Dateisystems, zum Beispiel unter /media/dvd, /media/usb,…). Inzwischen ist auch unter Linux ein „Hotplugging" möglich, bei dem das Montieren durch das Einstecken/Anschließen des Datenträgers automatisch ausgelöst wird.

Windows macht das „Einhängen" automatisch und blendet das Dateisystem unter einem Laufwerkbuchstaben (zum Beispiel E:, F:,…) ein. Das „Herausziehen" des Datenträgers unter Windows sollte wie bei Linux vorher durch die Funktion *„Hardware sicher entfernen"* angekündigt werden, sonst sind Datenverluste vorprogrammiert. Da Windows jederzeit mit dem unangekündigten „Herausziehen" des Datenträgers rechnen muss, kann es meist nicht so ausgeklügelte Chaching/Pufferungs-Mechanismen wie Linux einsetzen.

Unter Windows gibt es nur wenige Gerätedateien (zum Beispiel COM1, COM2, COM3, COM4, CON, LPT, LPT1, LPT2, LPT3, AUX, NUL) um die Serielle Schnittstelle *(communication)*, die Konsole *(console)*, die Drucker *(line printer)* und die serielle Schnittstelle *(auxiliary)* anzusprechen oder beliebige Daten wegzuwerfen *(null)*.

Der Zugriff auf Geräte wird nicht über Dateirechte geregelt, sondern über den HAL *(Hardware Abstraction Layer)*.

Wenn Du in Linux nicht der Gruppe „audio" angehörst, wirst Du auch keine Musik hören können. Ist so. Alles was nicht ausdrücklich erlaubt ist, ist verboten.

Keine Angst, bei den großen Linux-Distributionen ist das alles natürlich schon benutzerfreundlich vorkonfiguriert. Da brauchst Du in der Regel nie was ändern.

Ein Nachteil der kommando-orientierten Bedienung ist die längere „Anlaufzeit“, bis man damit gut umgehen kann. Mit einer grafischen Oberfläche kann man dagegen in der Regel sofort „loslegen“.

Grafische Oberflächen sind natürlich auch unter Linux verfügbar. Der wesentliche Unterschied liegt aber darin, dass diese Oberfläche (meist das X Window-System) kein fester Bestandteil des Betriebssystems ist, sondern wie jedes andere Anwender-Programm auch gestartet wird (ohne Administrator-Rechte), d. h. Fehler darin oder ein Absturz der grafischen Oberfläche haben keine Auswirkung auf das eigentliche Betriebssystem (wie bei Windows).

Die Entkopplung von Betriebssystem und GUI ist auch der Hauptgrund für die einfache Remote-Anwendung und -Administration von Linux.

Unter Linux gibt es ebenfalls Benutzer-Gruppen. Darin kann man mehrere Anwender zusammenfassen und anschließend Dateien für diese Gruppen mit Rechten versehen. Somit spart man sich die Arbeit, für jeden einzelnen Anwender eine Datei mit Zugriffsrechten zu versehen. In großen Netzwerken mit vielen Anwendern erleichtert das die Verwaltung erheblich.

Allerdings ist in einer Gruppe keine Zusammenfassung von anderen Gruppen möglich (wie bei Windows).

In mindestens einer Gruppe, der sogenannten *„Primären Gruppe“* (heißt meist „users" oder „wheel") ist man immer Mitglied. Neu angelegte Dateien und Verzeichnisse werden automatisch dieser Gruppe zugeordnet.

Unter Windows NT/2000/2003/2008/XP/Vista/7 existiert pro Anwender ein Verzeichnis *„Eigene Dateien“*, in dem er standardmäßig seine Dateien ablegt. Allerdings hat man als Anwender an vielen anderen Stellen im Windows-Dateisystem ebenfalls Schreibrecht und muss daher seine Dateien nicht unbedingt darin ablegen. Eine besonders beliebte, aber auch unschöne Art der Dateiablage ist der *„Desktop“*, wo die Dateien dann als *„Icons“* herumliegen und das Benutzer-Profil anwachsen lassen.

Unter Windows gibt es also viele mögliche Ablageorte für die Dateien eines Anwenders, dies verwirrt gelegentlich den Benutzer:

- Eigene Dateien
- Desktop
- Lokale Laufwerke (C:, D:)
- Netzwerk-Laufwerke

Er kann diese meist nicht klar unterscheiden (zum Beispiel *„wird das jetzt mitgesichert oder nicht?“*). Auf dem Desktop abgelegte Dateien blasen auch das *„Profil“* zu einer großen Datei auf, die das An- und Abmelden stark verlangsamt. Besonders auf dem Server verwaltete *„Roaming Profiles“* erleiden hierdurch massive Performance-Einbußen, aber auch bei lokal verwalteten Profilen ist so etwas durchaus spürbar.

Unter Linux existiert für jeden Anwender ein sogenanntes Heimatverzeichnis für seine persönlichen Daten. Nach der Anmeldung springt er sich automatisch dorthin. Dieses Verzeichnis gehört ihm, und er kann darin (fast) alles tun und lassen, was er will: Seine Daten speichern, seine E-Mails archivieren und eigene Programme installieren. Auch die individuellen Einstellungen für seinen Desktop und der von ihm eingesetzten Programme werden dort gespeichert. Auf alle anderen Stellen des Dateisystems hat man als Anwender keinen schreibenden Zugriff (außer im temporären Verzeichnis „/tmp").

Dieses Heimatverzeichnis ist ein privater Bereich, und alle Konfigurationen, die die persönliche Arbeitsumgebung betreffen, werden hier gesammelt und gespeichert. Diese Daten werden weder vom Betriebssystem noch von anderen Anwendern verwendet oder geändert.

Ein wichtiger Unterschied zwischen Linux und Windows besteht auch in der Bedeutung der Kommandozeile. In den Zeiten von MS-DOS wurden Befehle am DOS-Prompt eingetippt und ausgeführt. Mit Windows hat sich dies grundlegend geändert. Fast alle Programme werden in der grafischen Oberfläche mittels Mausklick ausgeführt. Es existiert zwar noch ein DOS- oder CMD-Fenster, doch dieses hat fast keine Bedeutung mehr.

Unter Linux stellt die Kommandozeile unter der Shell in einem Terminal nach wie vor ein sehr wichtiges Arbeitsmittel dar. Alle Programme können von dort aus gestartet werden (wobei grafische Anwendungen natürlich eine X Window-Umgebung für den Start benötigen) und die komplette System-Administration kann darüber durchgeführt werden. Dadurch dass keine Gui „mitgeschleppt“ werden muss, eignet sich diese Methode der Systemwartung auch sehr gut für Remote-Wartungen mit keiner performanten Infrastruktur (Modemleitungen...)

Die unter Linux verfügbaren Shells sind deutlich mächtiger als die Shell des Kommandozeilenfensters unter Windows. Sie eignen sich auch zur Programmierung und automatisierten Ausführung von Tätigkeiten.

Leider gibt es verschiedene Shells (zum Beispiel sh, csh, tcsh, ksh, bash, zsh), die sich in Syntax und Funktionalität unterscheiden. Unter Linux hat sich aber inzwischen mit der sehr leistungsfähigen bash eine Standard-Shell herauskristallisiert.

Windows war und ist primär ein Einzelbenutzer-System, auch wenn Windows NT/2000/2003/-2008/XP/Vista/7 inzwischen echte Mehrbenutzerfähigkeit auf Prozessebene anbietet. Es gibt aber kaum Mechanismen, um auf einer entfernten Windows-Maschine komfortabel arbeiten zu können.

Diese Lücke wird zwar von den *„Windows Terminal-Servern“* (zum Beispiel Citrix) geschlossen, für die jedoch die Verwendung spezieller Clients notwendig ist und die zusätzliche hohe Lizenzgebühren kosten. Alle Windows-Server-Versionen enthalten deshalb bereits einen Windows Terminal-Server mit zwei Client-Lizenzen.

Zugriffsrechte

Windows 95-ME

(Sorry, muss angeschnitten werden zwecks Verständnis). Bei Windows 95/98/98SE/ME kann man als normaler Anwender noch jede Datei auf dem System ansehen, verändern, speichern und löschen, es gibt keinen Datei-Besitzer und keine Zugriffsrechte. Dies hat seine Ursache im dort verwendeten FAT-Dateisystem, das keine Zugriffsrechte und Besitzverhältnisse kennt.

Die Anmeldung dient nur zur Auswahl des passenden Anmelde-Profils (speichert pro Anwender lediglich seine persönlichen Einstellungen und den eingerichteten Desktop), das Passwort ist hingegen beliebig wählbar (wird aber bei Anbindung an Netzwerk-Freigaben verwendet).

Pro Datei gibt es immerhin folgende fünf „Datei-Attribute“, die schon in MS-DOS vorhanden waren:

Attribut	Bedeutung	
ARCH	*Archive*	Datei seit der letzten Sicherung geändert
HID	*Hidden*	Versteckte Datei (wird nicht aufgelistet)
RO	*Readonly*	Nur lesbar
SYS	*System*	Systemdatei
VOL	*Volume*	Datenträger-Bezeichnung (1x pro Partition)

Windows NT-XP

Die Versionen Windows NT/2000/2003/2008/XP/Vista/7 kennen darüber hinaus aufgrund des meist verwendeten NTFS-Dateisystems Besitzverhältnisse folgende 6 Zugriffsrechte:

Recht	Bedeutung	
R	*read*	Lesen
W	*write*	Schreiben
X	*execute*	Ausführen
D	*delete*	Löschen
P	*change permission*	Berechtigung ändern
O	*take ownership*	Besitz übernehmen

Diese Basisrechte werden zur einfacheren Rechtevergabe und Rechteanzeige zu folgenden 7 Gruppen zusammengefasst (für Verzeichnisse gibt es alle 7 Gruppen, für Dateien nur 4 Gruppen):

Recht	Bedeutung	Datei	Verz.
------	no access	Kein Zugriff	— ja
R-----	list	Anzeigen ja	ja
R-X---	read	Lesen ja	ja
WX---	add	Hinzufügen	— ja
RWX---	read + add —	Lesen + Hinzufügen j	
RWXD--	change	Ändern ja	ja
RWXDPO	all ja	Vollzugriff ja	

Besitzer(Gruppe)

Jede Datei und jedes Verzeichnis hat einen Besitzer *(Owner)*, der normalerweise als einziger berechtigt ist, die Zugriffsrechte einer Datei zu ändern. Durch das Recht P = *„Berechtigung übernehmen“* kann er dieses Recht auch an andere Benutzer/Gruppen weitergeben.

Vererbung

Verzeichnisse haben unter Windows NT/2000/2003/2008/XP/Vista/7 zwei Rechtesätze. Der erste gilt für das Verzeichnis selbst, der zweite gilt für alle darin neu erzeugten Objekte (Dateien und Verzeichnisse), d. h. darüber wird eine „Vererbung“ der Dateirechte realisiert. Zwischen WinNT und Win2000 hat sich die Semantik dieser Vererbung geändert. Bei WinNT wurde statisch vererbt, ab Win2000 wird dynamisch vererbt. D. h. Änderungen an Zugriffsrechten eines Verzeichnisses wirken sich ab Win2000 dynamisch auf alle seine Unterdateien und Unterverzeichnisse aus.

Kombination

Die Rechtesätze können in den zu jeder Datei/jedem Verzeichnis vorhandenen ACLs *(Access Control Lists)* für beliebige Benutzer und Benutzer-Gruppen getrennt vergeben werden. Es ist also eine sehr „feingranulare“ Rechtevergabe

möglich, deren Feinheiten aber erst einmal verstanden werden müssen. Keine Angst, nach zwei Jahren hast Du's dann auch drauf :-)

Die obigen 6/7 Rechte können in jeder ACL gesetzt oder gelöscht werden. Ist ein Benutzer Mitglied in mehreren Gruppen, für die zu einer Datei oder einem Verzeichnis eine ACL vorhanden ist, so werden die gesetzten Rechte dieser ACLs verodert (die Obermenge gilt) und mit den gelöschten Rechten dieser ACLs verundet (die Untermenge gilt). D. h. durch Hinzufügen von ACLs zu einem Dateisystemobjekt können Zugriffsrechte auch wieder weggenommen werden.

Ehrlicherweise brauchst Du das nicht wirklich verstehen. Du solltest nur wahrnehmen, dass die Rechteverwaltung in Windows nicht ganz trivial ist. Viel Übung, Lernaufwand und Geduld sind für deren Beherrschung allemal nötig. Auf Holzschlapfen drückst die auch nicht mehr so locker runter. Die Windows98-Zeiten sind definitiv vorbei.

Linux

Dateirechte

Linux kennt an Zugriffsrechten für jede Datei die drei Rechte

Recht	Bedeutung	
r	read	Lesen
w	write	Schreiben
x	execute	Ausführen

Die drei Rechte regeln, was mit dem Datei-Inhalt erlaubt ist:

Recht	Bedeutung	
r	read	Lesen des Datei-Inhalts
w	write	Schreiben des Datei-Inhalts
x	execute	Ausführen des Datei-Inhalts als Programm

Diese drei Rechte gibt es jeweils für die drei Benutzertypen (also insgesamt 3 × 3 = 9 Rechte):

Typ	Bedeutung	
u	user	Besitzer
g	group	Besitzer-Gruppe
o	other	Alle Anderen

Will man alle drei Benutzertypen zusammen ansprechen, ist statt ugo das Kürzel a (all) verwendbar. Diese neun Rechte werden mit dem Kommando ls -l (long) immer in der gleichen Reihenfolge in der ersten Spalte aufgelistet (fehlt ein Recht, so wird stattdessen ein „-" angezeigt):

User	Group	Other
r w x	rwx	rwx

Die gleichen drei Rechte sind auch für Verzeichnisse vorhanden und regeln dort, was mit den Dateinamen im Verzeichnis erlaubt ist:

Recht	Bedeutung	
r	read	Auflisten des Inhalts (ls)
w	write	Anlegen, Umbenennen, Verschieben, Löschen von Dateien/Verz. darin (touch mkdir mv rm rmdir)
x	execute	Betreten des Verzeichnisses (cd)

Das x-Recht eines Verzeichnisses entscheidet nicht nur über den Zugang zu diesem Verzeichnis, sondern auch über den Zugang zu allen seinen Unterverzeichnissen und ihren Dateien (sofern keine Hard Links auf Dateien eingesetzt werden).

Rechteanomalie

Die Definition der Zugriffsrechte scheint zunächst eine sogenannte Rechteanomalie zu bedingen: eine Datei kann gelöscht werden (Schreibrecht in ihrem Verzeichnis), obwohl sie nicht schreibbar ist (Schreibrecht für die Datei selbst). Dabei handelt es sich aber nur um eine streng logische Definition und Anwendung dieser Rechte. Mit dem Sticky-Recht kann diese Rechteanomalie bei Bedarf „behoben" werden. (Sticky-Bit lass ich einmal aus… ist ein Spezialthema und im Heimuserbereich nicht wirklich ein Thema…)

Besitzer(Gruppe)

Besitzer einer Datei/eines Verzeichnisses ist anfangs derjenige, der das Objekt anlegt, Besitzer-Gruppe ist anfangs seine Primäre Gruppe (meist „user"). *„Alle Anderen"* umfasst die restlichen Anwender und Gruppen.

Die Besitzer-Gruppe kann jederzeit vom Besitzer geändert werden. Der Besitzer kann nur vom Administrator geändert werden! Die Zugriffsrechte kann nur der Besitzer (oder der Administrator) ändern. Im Unterschied zu Windows gibt es also kein Recht, das einem die Besitzübernahme einer Datei oder die Änderung von Zugriffsrechten gestattet!

Da (nur) der Besitzer einer Datei immer ihre Zugriffsrechte ändern kann, kann er sich bei Bedarf immer die von ihm benötigten Zugriffsrechte daran verschaffen.

Wird eine Datei/ein Verzeichnis neu angelegt, so gehört sie/es automatisch dem angemeldeten Benutzer und seiner aktuellen Gruppe (meist users oder staff). Als Standard-Zugriffsrechte werden die maximal sinnvollen eingestellt, d. h.

Typ	Rechte	
Datei	666	rw-rw-rw-
Verzeichnis	777	rwxrwxrwx

Begründung: Dateien sollen nur ganz selten ausführbar sein. Wird dies gewünscht, dann ist es mit dem Befehl chmod a+x DATEI explizit einzustellen. Verzeichnisse sollen immer ausführbar sein, sonst kann man nicht mit cd in sie hinein wechseln und nicht mit ls ihren Inhalt auflisten.

Mit dem Befehl umask können von diesen maximalen Rechten einige entfernt werden, während eine Datei oder ein Verzeichnis angelegt wird (für bereits angelegte Dateien/Verzeichnisse hat dieser Befehl keine Wirkung!). Normalerweise wird diese sogenannte Usage-Mask der zu entfernenden Rechte in Oktalform angegeben (genau deshalb muss man sich mit dieser Oktalform überhaupt auseinandersetzen). Hier einige umask-Werte und die daraus resultierenden Standardrechte:

Usage-Mask	Resultierende Rechte	
	Datei	Verzeichnis
000	rw-rw-rw-	rwxrwxrwx
002	rw-rw-r--	rwxrwxr-x
022	rw-r--r--	rwxr-xr-x
007	rw-rw----	rwxrwx---
027	rw-r-----	rwxr-x---
077	rw-------	rwx------
277	r--------	r-x------
777	---------	---------

Unter Linux findet (bis auf eine Ausnahme: Set-GroupID-Recht) keine Vererbung der 12 Standard-Rechte statt. Jedes Verzeichnis und jede Datei hat ihren eigenen Standard-Rechte-Satz. Das Set-GroupID-Recht bei einem Verzeichnis sorgt dafür, dass dieses Recht selbst und die Besitzer-Gruppe beim Anlegen eines neuen Verzeichnisses darin an dieses vererbt wird.

Der zweite ACL-Rechtesatz von Verzeichnissen wird beim neu Anlegen von Dateien und Verzeichnissen darin an diese als Standard-Rechtesatz und als Default-Rechtesatz vererbt.

Durch Änderung der Zugriffsrechte eines Verzeichnisses kann der Zugang auf den kompletten Dateibaum darunter beeinflusst (erlaubt/gesperrt) werden (sofern keine Hard Links von außen auf Dateien in diesem Dateibaum zeigen).

Komplexität

Durch die bei Windows üblichen ACLs auf Gruppen und Benutzer wird die Anzahl der Rechte-Kombinationen sehr hoch, mit entsprechenden Folgen für die Übersicht und Verständlichkeit.

Bei Verzicht auf ACLs bleibt das Linux-Rechtemodell sehr einfach. Aber auch wenn ACLs verwendet werden, ist die Verständlichkeit der möglichen Rechtekombinationen einfach

Da die beschriebenen Zugriffsrechte und der Datei-Besitzer bzw. die Datei-Besitzer-Gruppe unter Linux von Beginn an vorhanden waren, sind dort alle Anwendungen daran angepasst. Das Linux-Standard-Zugriffsrechte-System ist relativ einfach, ACLs sind inzwischen möglich, werden aber selten eingesetzt. Im Heimbereich praktisch nie.

Die unter Windows NT/2000/2003/2008/XP/Vista/7 vorhandenen feingranularen Möglichkeiten zur Zugriffsrechte-Steuerung werden aufgrund der Windows-Historie von den Windows-Anwendungen zu wenig berücksichtigt. Daher gibt es auch heute noch Windows-Anwendungen, die Zugriffsrechte nicht beachten und nur mit Administrator-Rechten laufen (zum Beispiel Spiele). Viele ältere Anwendungen und sogar neue Software wurden nie dafür entwickelt, ohne vollständige Administratorrechte zu laufen. Lässt man diese Software mit eingeschränkten Rechten laufen, treten Fehler auf oder die Software arbeitet nicht ordnungsgemäß. Das ist eben das XP-Erbe, das meist einen Neukauf der „alten" (unter XP) tadellos gelaufenen Software nach sich ziehen kann. Das Windows-Zugriffsrechte-System ist sehr komplex. Funktioniert auch in einer Firmenumgebung sehr gut. Das steht einmal fest.

Zugriffsrechte-Systeme sollten einfach sein, damit Anwender sie auch verstehen können, nur dann werden sie (vernünftig) eingesetzt. Sie sollten aber auch leistungsfähig genug sein, um die wichtigen Anwendungsfälle abzudecken. Das Linux-Standard-Rechtesystem (ohne ACLs) erfüllt diese beiden Kriterien. Bei ACLs sind die Meinungen geteilt, auch wenn Windows-Kenner es für das Non-plus-ultra halten und vom Standard-Linux-Rechtesystem erst einmal enttäuscht sind.

„Meine Eindrücke"

Die „Probleme" werden nicht kleiner werden. Speziell im Heimuserbereich, wenn der HobbyAdmin versucht, auf seinem NAS zu Hause die Rechte zu setzen, damit er mit Windows7 darauf zugreifen kann. Die meisten NAS-Geräte laufen aber unter Linux und demzufolge auch mit deren Rechteverwaltung. Da brauchst du dann erst recht zumindest Basiskenntnisse der Linux-Rechteverwaltung. Wer hat die schon? Der nette Bekannte, der immer mit neuen Tuningtools vorbeikommt und die Kiste zuinstal-

liert? Abgesehen davon, weiß der überhaupt was er da am System macht? Primär ist ein Tuningtool als zusätzliche Fehlerquelle zu betrachten.

Ich habe Leute gesehen, die haben vier zusätzliche Searchbars beim Internetexplorer oben dranhängen. *„…keine Ahnung, wie die raufgekommen sind…“*.

Das ist leider noch immer weit verbreitet. Nicht die Searchbars meine ich jetzt (die auch). Aber das automatisierte Abnicken von sämtlichen Dialogboxen, die da auftauchen *(…wird scho passn…)*.

Sei es jetzt ein Update für eine Java-Maschine, die Installation eines Bildschirmschoners oder eine Systemwarnmeldung, dass der Speicherplatz knapp wird.

Die meisten lesen nicht einmal das, was sie abgenickt haben. Jetzt kannst Du Dir vorstellen, was sich da im erhöhten Modus (Administrator) abspielt. Also, den Windows-Supporter wird's freuen. Ohne Zweifel.

Ich sag immer „meinen Leuten“, die Windows verwenden: Wenn was kommt, lesen und im Zweifel immer verneinen. Unter Linux ist es egal. Die haben eh kein root-Passwort. Da gibt's die wenigsten Brösel. Das kannst Du bei einem Windows-User nicht machen. Der braucht unbedingt sein Admin-Passwort. Für was auch immer. Das bekommst Du schwer aus den Köpfen raus. Ist so.

Eine Korrektur noch. Die Eingabeaufforderung unter Windows7 im erhöhtem Modus hat doch eine visuelle Unterscheidung. Am Anfang ist immer das Präfix *„Administrator“* ersichtlich. Sorry, mein Fehler.

Generell ist der Windows-Weg mit der Rechteverwaltung zu begrüßen. Die Philosophie von Linux finde ich aber trotzdem „besser“. Nämlich alles einmal zu verbieten, was nicht explizit erlaubt ist. *Wenn'st was willst, musst es selber „aufdrehen“.*

Windows versucht diesen Gedankengang immer irgendwie „aufzuweichen“ (granulare UAC, kein Vollzugriff auf Systemfiles als Admin…). Das verkompliziert das System aber nur noch mehr. Bei den meisten Linux-Distributionen wird demgegenüber das KISS-Prinzip angewendet.

KISS= *keep it simple and stupid* (halt es so einfach wie möglich)

Erschwerend kommt natürlich die Haltung der meisten Privatuser hinzu. *(…bin e alleine am PC… i tua jo nur Büdln schaun und a bissl sörfn… do brauch i jo kane Rechte… schoit ma de UAC afoch ob… tua ma des mochn, dass i orbeitn kann…)*

Wie man es einführt, so hat man's eben. In Linux brauchst Du in dieser Hinsicht keine Überzeugungsarbeit leisten. Dort war das schon immer so mit den Rechten. Deshalb wurde es auch immer von Windows-Usern als „kompliziert“ angesehen.

Wie schon einmal geschrieben, die Akzeptanz der User wird da die größte Hürde sein.

Natürlich kannst Du extra gewisse Anwendungen auch von der UAC „befreien“. Du musst nur wissen, was Du tust.

„Kommandozeile“

Wer mit Linux vor allem als Anfänger arbeitet, der versucht einmal die Kommandozeile zu vermeiden. War bei mir auch so. Sobald ich das Wort schon hörte, manifestierten sich bei mir Gedanken wie untersetzter, vereinsamter Gurus, die hinter zentimeterdicken Brillengläsern ihre kryptischen Zeichen auf schwarzem Hintergrund eingaben.

Das erklärt vielleicht auch den Erfolg von Ubuntu-Linux. Als normaler User wirst Du dort nur in Ausnahmefällen auf die Kommandozeile gebeten; wenn überhaupt.

PS: shell, Terminal, Kommandozeile, Konsole, Eingabeaufforderung… ist alles das selbe.

Alles ist schön per Mausklick konfigurierbar (oder fast alles). Das ist auch gut so und hilft der Popularität von Linux weiter.

Die grafischen Lösungen haben aber immer einen Haken. Sie sind begrenzt in ihrem Funktionsumfang. Gilt für Linux und Windows.

Nimm einfach eine Windows-Homeversion her. Wie aktivierst du da auf die Schnelle den Administrator auf der GUI?

(GUI= grafisches User Interface)

Frag' mich nicht, keine Ahnung, ehrlich. Aber ich weiß, wie ich in Windows auf der Kommandozeile diesen Kollegen freischalte.

Wie prüfst Du dein Netzwerk? Richtig mit „ping“ und „tracert“ auf der Kommandozeile.

Obwohl Windows ein rein grafisches System ist, kannst Du manche Sachen nur auf der Eingabeaufforderung bewerkstelligen.

Das ist ja nichts Schlechtes. Aber wie bei Linux so hat auch in Windows diese Arbeitsweise einen unschätzbaren Vorteil. Du kommunizierst direkt mit der Hardware (oder dem Kernel).

Somit umgehst Du das ganze grafische Geraffel mit seinen eventuellen Fehlerquellen. Außerdem hast Du im Gegensatz zu sehr bedingt aussagekräftigen Dialogboxen *(pfau, da Boikn hängt jetzta scho ganz sche lang…)* auf der Kommandozeile immer glasklare Statusmeldungen. In Textform.

Voraussetzung: Du musst lesen können.

Nachteil: Du brauchst natürlich Einarbeitungszeit für die Syntax der Befehle.

Wenn Du systemrelevante Aktionen erfolgreich durchführen willst, ist die Beherrschung der Kommandozeile absolute Pflicht. Gilt für Windows und Linux.

Die Kommandozeile hat vor allem in Linux noch einen entscheidenden Vorteil. Sie funktioniert immer. Das Erste, was ein Linux-Admin bei einem Problem immer macht. Er schaut, dass er eine Shell bekommt. Von dort hat er überall Zugriff auf das System und Ressourcen. Den interessiert meist gar nicht, was da auf der Grafik passiert (oder nicht passiert) ist. Er sieht's sowieso in der Shell.

Alles, was Du in Linux per Dialogbox ankreuzerlst oder anklickst, wird letztendlich in Textform in ein File geschrieben. Und das File sieht eh der Administrator. Also warum da umständlich mit den Menüs herumklicken, wenn man das direkt im Konfigurationsfile fixen kann.

Ein kleines Beispiel: Wie heißt mein „hostname“?

Ich gebe den Befehl ein:

```
cat /etc/hostname < Befehl
antichrist < Egebnis...mein Rechnername
```

das war's. Ich wüßte jetzt ehrlich auch nicht, wo ich da ad hoc in der GUI nachschauen sollte. Nebenbei, diese „Kontrollzentren“ à la Windows gibt's hauptsächlich bei den einsteigerfreundlichen Linux-Versionen (Ubuntu, Suse...).

Die Konsole gibt's aber überall. Die reicht auch in der Regel. Was sollte ein User auch im Kontrollzentrum wollen? Hat sowieso keine Rechte dort. Und als Admin kommst über kurz oder lang um die Konsole sowieso nicht herum.

Ein weiterer Grund ist die Vielzahl an verschiedenen grafischen Desktops unter Linux (KDE, Gnome, Xfce, Lxde, Fluxbox...), um nur die bekanntesten zu nennen. Glaubst Du wirklich, dass jeder Linux-Admin die Dialogboxen von verschiedenen Desktops auswendig nachvollziehen kann? Unmöglich, es gibt zu viele Möglichkeiten. Aber nur eine auf der Kommandozeile.

Ein erfahrener Windows-Admin setzt Dir ein albanisches Windows genauso schnell auf wie ein Deutsches. Weil er eben die Dialogboxen und Menüs schon intus hat. In Windows gibt's nur einen Desktop. Und der schaut in Deutschland genauso aus wie in Italien, Holland oder Japan.

Kannst Du bei Linux vergessen. Dort gibt's zu viele Möglichkeiten der Desktopgestaltung.

Noch ein Beispiel: Wenn Du in Linux-Foren um Hilfe schreist, schicken Dich alle einmal auf die Kommandozeile, um irgendwelche Befehle einzutippen und dessen Ergebnisse zu posten. Warum? Weil es einfach am aussagekräftigsten in Textform ist. Außerdem braucht man den User nicht durch den Urwald an Dialogboxen und Dropdownmenüs durchzulotsen, falls das überhaupt funktioniert.

Wie setzen Webmaster einen Linux-Server auf? Und vor allem die Ordnerstruktur in einem Webserver? In Windows wirst Du Dir halt einen Wolf klicken. In Linux machst Du das auf der Kommandozeile: **siehe Bild 33**

Sieh auf die markierte Zeile > mkdir -p 1/2/3/4 5 6 7 8

mkdir= make directory = mach einen Ordner

-p = make parents =mach auch Unterverzeichnisse

Der obere Befehl führt demnach zu folgender Ordnerstruktur: **siehe Bild 34**

Sprich, mit einem Einzeiler auf der Kommandozeile kann ich eine bestimmte Ordnerstruktur einfach realisieren, ohne mich da jetzt grafisch durch jede Ebene hindurchzuklicken und diese zu bearbeiten.

Das war einmal nur ein triviales Beispiel für die Anwendung der Shell.

Meine Erfahrungen

Wenn Du eine Aufgabe nur einmal im Jahr machst, ist meist die grafische Lösung zu bevorzugen, weil Du Dich nicht an die entsprechenden Befehle auf der Kommandozeile „erinnern“ musst.

Für automatisierte Aufgaben, Reparaturen oder Wartungszwecke ist meist die Konsole besser geeignet.

„Meine“ Linux-User kommen auch nie mit der Konsole in Berührung. Warum auch? Kommen die Firmenmitarbeiter mit der Eingabeaufforderung in Windows XP in Berührung? Eben.

Daher: Kommandozeile sollte „nur“ dem Systemverwalter zugemutet werden. Gilt für Linux und Windows.

Da im privaten Umfeld die meisten aber gleichzeitig „Administrator“ sind, gibt es eben oft Berührungspunkte mit der Eingabeaufforderung.

Lass' Dir auch gesagt sein, dass speziell in Linux die Benutzung der Konsole aus rein praktischen Gründen geschieht. Wenn wer damit angeben will… ja, wenn's ihm hilft.

Du wirst auch mit der Zeit draufkommen, dass die Konsole in Linux ein extrem mächtiges Tool

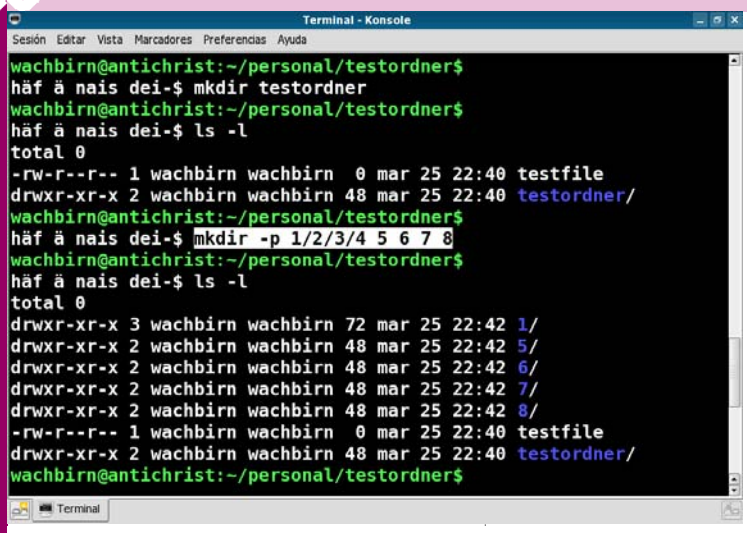


Bild 33: Ordner anlegen

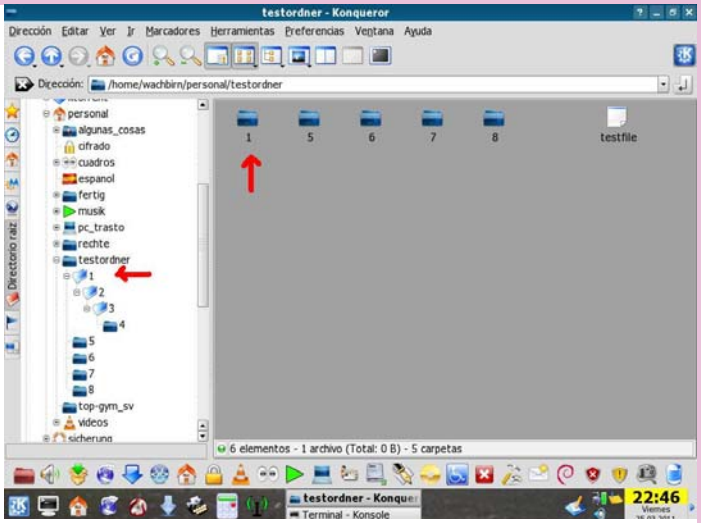


Bild 34: Ordnerstruktur

ist. Nicht einmal annähernd mit der Windows-Eingabeaufforderung zu vergleichen. Zusätzlich gibt es aber jetzt die Windows-Shell, mit der ich mich aber ehrlicherweise noch nicht genug beschäftigt habe.

Aber alleine die Tatsache der Einführung oben zitierter Windows-Shell sagt mir, dass auch Windows dieses Tool zukünftig nicht missen möchte.

Leider ist die Syntax wieder eine eigene und ein großer Einarbeitungsaufwand auch in Windows für die Beherrschung dieser Shell notwendig. Die Schulungscenter müssen ja auch von was leben...

Somit kannst Du auch die Rechteverwaltung sehr effizient unter der Konsole anwenden. Klar wirst Du anfangs Fehler machen. Na und? Wie oft hast du dich schon in Windows verklickt oder in den Menüs verirrt?.

Lass dich nicht entmutigen. Egal, welches Linux du dann verwendest. Diese Befehle kannst du auf jedem Linux anwenden. Sogar bei Apple. Die Anwendung der Tools auf verschiedenen Plattformen ist ein enormer Gewinn.

Somit kommen wir zu einem weiteren wichtigen Punkt der Linux-Tools. Diese Tools sind in der Regel nur für einen bestimmten Zweck ausgerichtet und können meist nur eine bestimmte Aufgabe übernehmen. Diese aber ausgesprochen gut.

Durch Verknüpfung der verschiedenen Tools mit anderen hat man aber ein Baukastensystem, das sehr mächtig ist.

Bsp: Befehl> less textfile | grep finde_mich

less = ein Pager, der mir die Datei „Textfile“ darstellt

| = Ergebnis des ersten Befehles (less Textfile) an nächsten Befehl übergeben

grep = suche Ausdruck „finde_mich“ (in Textfile)

Durch die Verknüpfung von 2 Befehlen (less+-grep) habe ich ein Ergebnis bekommen. Alleine sind die Befehle sehr gut, aber nur in Kombination zueinander in diesem Fall sehr gut einsetzbar.

Erwartungshaltungen an Linux

„Wenn Linux genauso funktionieren würde wie Windows, hätte es viel mehr Akzeptanz und User...“ tönt es meist aus den Windows-Foren.

Zur Klarstellung: Wenn Linux wie Windows funktionieren würde, wäre es nur ein besserer Windows-Klon. Wer braucht so was?

Wenn ich so was will, kauf` ich mir gleich Windows 7.

Ist die selbe Geschichte wie damals, als der Firefox herausgekommen ist. Warum war (und ist) der so erfolgreich. Ganz einfach. Er hat nicht so funktioniert (Tabs, Suchmaschinenfeld...) wie der Internetexplorer.

Ob Linux mehr Akzeptanz und User hätte? Welchen Linux-User interessiert das wirklich? Vor allem, was würde es bringen? Okay, für aktuelle Hardwareunterstützung wäre es nicht so schlecht, damit die Hardwarehersteller Linux noch mehr Treibersupport zukommen lassen würden.

Mir raubt es jetzt aber nicht unbedingt den Schlaf, dass Linux weltweit bei knapp 2% Desktopanteil herumkrebst. Selbst wenn`s jetzt 70% wären, was würde ich dann mit dieser Aussage anfangen? Mich besser fühlen? Und vor allem, was würde es Linux bringen? Linux verdient keinen Cent mehr, wenn die Linuxuserzahl ansteigt. Linux selbst verdient überhaupt nichts. Gar nichts. Linux ist keine Firma. Deshalb sind diese Popularitätsvergleiche der Betriebssysteme grundsätzlich entbehrlich.

So überheblich es auch klingt. Ob das Zeugl jetzt 3 Milliarden verwenden oder 24 Leute. Ist mir wirklich egal. Es ist auf keinen Fall ein Entscheidungskriterium für mich. Du kaufst ja auch nicht das selbe Auto wie Dein Nachbar, oder?

Eines darfst Du vor allem nie vergessen: Linux ist keine Firma. Microsoft „muss“ hingegen Geld mit ihren Produkten verdienen.

In jedem Land hast Du eine Microsoft-Niederlassung. Hast du schon mal wo eine Linux -Niederlassung gesehen? Ich auch nicht. Es gibt kein Firmengebäude auf der ganzen Welt, wo oben in breiten Lettern „Linux“ draufsteht.

Linux ist freie Software. Microsoft ist eine „Firma“, die Produkte herstellt. Die 2 Sachen sind nicht wirklich vergleichbar. Wenn überhaupt, dann vielleicht deren Endprodukte.

Linux wird aber von großen Firmen unterstützt (IBM, Google...).

Klar verdienen viele Linux-Firmen auch Kohle mit Linux (Novell-Suse, Canonical-Ubuntu, Red hat...). Hauptsächlich durch Support.

Die Hauptintention bei Linux ist aber nach wie vor die Bereitstellung eines funktionierenden Betriebssystems. Mit unzähligen Projekten und Distributionen weltweit. Es soll einfach funktionieren. Der gewöhnliche Linux-User benutzt es nicht, weil‘s gratis ist, sondern weil es funktioniert. Was nützt ein Gratis-Betriebssystem, wenn‘st 3 mal am Tag abstürzt mit dem Dreck.

Generell wird die Betriebssystemwahl von vielen überschätzt. Bedenke, dass die meisten User „nur“ Anwender mit allerhöchstens rudimentären Systemkenntnissen sind. Da ist es eigentlich egal, welche Plattform sie benutzen. Funktionieren muss es. Kein solcher User sollte je in die Verlegenheit kommen, an den Berechtigungen herumdoktern zu müssen.

Und ja, die Rechteverwaltung wirkt in Linux sehr antiquiert. Man kann es auch „ausgereift“ nennen. Das Telefonnetz der USA läuft großteils noch unter Unix und deren Rechte- und Userverwaltung. Wenn‘s funktioniert, warum nicht?

Linux Grundsatz: *If its aint broken, dont fix it. Wenns nicht hin ist, laß‘ es in Ruhe.*

Das ist eines der Dinge, die mir in Linux sehr gefallen. Wenn was gut funktioniert, läßt man es und doktert nicht herum. Das Wissen in der Rechteverwaltung kann ich im privaten Umfeld seit Jahren unverändert anwenden.

Frag‘ mich mal, was ich da seit Windows2000 alles „büffeln“ hab können, damit ich das auf der Windows-Client Seite schnall? Und was sich da mit jeder Version (W2k/XP/Vista/7...) geändert hat. Und erst mit den verschiedenen Auslieferungen (Home, Starter, Professionell, Ultimate....).

Die Windows-Servergeschichten noch gar nicht eingerechnet...

Keine Ahnung, was sich bei Windows 8 alles noch ändern wird. Vor allem, warum ändern sie es dauernd? Weil eben der User schrittweise an das Optimum herangeführt werden muss (warum auch immer).

Bei Linux war er gleich dort. Klonk. (Da hast a Linux-CD. Drauf ist ein vollwertiges Multiuser- und Netzwerkbetriebssystem mit einer implementierten User- und Rechteverwaltung... viel Spaß)

Windows 7 ist zumindest in der Rechteverwaltung auf einem linux-ähnlichen Weg unterwegs.

Klar ist‘s ein Schock, erst einmal mit Verboten am eigenen System konfrontiert zu werden. Aber Windows hat dafür gut 7 Jahre gebraucht (W2k-Vista). Und vor allem bekommt es jetzt nur sehr schwer den XP-Standard aus den Köpfen *(i brauch de UAC net...)*.

Darum ist Linux auch „ganz anders“ als Windows. Es sind 2 Welten und Philosophien, die da aufeinanderprallen. Immer. Da kannst Du stundenlang fruchtlos diskutieren. Jeder wird eben nach seiner Präferenz argumentieren.

Und unser Bill Gates endlich seinen Linux-Server im Keller updaten :-)

Überlegungen UAC und Linux

Standardmäßig richtet Windows ein Benutzerkonto mit „quasi-administrativen“ Rechten ein.

In Linux hat das Benutzerkonto nur Userrechte. Also keinen Systemzugang. (wie mit der UAC bei Windows).

Ein Klick in Windows genügt, und die Prozesse erhalten Vollzugriff auf sicherheitsrelevante Sachen.

Von Microsoft zertifizierte Prozesse arbeiten sogar ohne zusätzlichen Klick.

Wem das zu unsicher ist, sollte (kann) sich ein Standardkonto anlegen. Geh unter Systemsteuerung einfach auf „Benutzerkonten“ und leg einen neuen Benutzer an. Wähle *„Standardbenutzer“* als Konotyp. Jetzt vergibst du noch ein Passwort. Fertig.

Dieses Konto könntest Du jetzt noch je nach Paranoiaausprägung über Gruppenrichtlinien oder den UAC-Schieberegler granular anpassen.

Pass auf, wenn Du den Schieberegler ganz nach unten gibst, verliert auch der Internet-Explorer seinen geschützten Modus. Nicht gut. Und vor allem solltest Du nie den Regler ganz nach unten fahren. Warum das MS überhaupt anbietet...?

Der andere Weg, der für die meisten User eher nachvollziehbar wäre, ist den installierten User abzusichern.

Als erstes könntest Du den Schieberegler ganz nach oben fahren. So vermeidest Du, dass sich Prozesse unbemerkt höhere Rechte verschaffen können. Die UAC warnt jetzt halt immer.

Was bleibt, ist dass ein einzelner Klick zur Ausführung von Anwendungen genügt. Das wollen wir nicht. Manchmal ist der Finger eben schneller als das Hirn.

Wir könnten zum Beispiel das Admin-Passwort einfordern.

Um das Administrator-Passwort bei jeder Authentifizierung einzufordern (in Windows), müssen wir in die Eingeweide von Windows hinabsteigen.

Sorry, für die niederwertigen Windows-Versionen sieht‘s da nicht so gut aus. Professional und Ultimate-User erledigen das bequem über die Sicherheitsrichtlinie.

Gehe ins Startmenü und gib im Suchfenster `"secpol.msc"` ein. Enter drücken. Die gesuchte Option verbirgt sich unter *„Lokale Richtlinien | Sicherheitsoptionen | Benutzerkontensteuerung: Verhalten der Benutzeraufforderung mit erhöhten Rechten für Administratoren im Administratorbestätigungsmodus“*.

Doppelklick auf den Eintrag und wähle die Einstellung *„Eingabeaufforderung zu Anmeldeinformationen auf dem sicheren Desktop“*.

Besitzern von kleineren Windows 7 Versionen steht das Snap-In *„Sicherheitsrichtlinien“* nicht zur Verfügung. Die müssen das über einen RegistryEingriff fixen. Hab ich aber noch nie gemacht, deshalb kommt da von mir auch nix mehr.

Die UAC ist kein Schutz vor Malware. Malware = *malicioussware* = schädliche Software.

Auch Microsoft distanziert sich von der UAC als Abwehrmechanismus gegen Malware.

Bei Vista mussten Anwender bis zu drei Dialoge absegnen, nur um einen einzelnen Kopiervorgang durchzuführen. Der belehrende Aspekt der UAC sorgte auf Anwenderseite daher eher für Ärger und nicht für mehr Sensibilität.

Warum Microsoft die erweiterten Rechte nicht konsequent und vor allem einmalig für die Erledigung einer administrativen Aufgabe einfordert, bleibt trotzdem unklar.

Mac OS und Linux-Nutzer haben sich an eine solche Authentifizierung längst gewöhnt. Gibst einmal das Admin-Passwort ein, und gut is.

Immerhin bietet Windows 7 jetzt die Möglichkeit, dieses Verhalten durch Veränderung der UAC-Standardeinstellungen zu erzwingen. So gesehen war Vista schon gut, da man jetzt den direkten Vergleich hat, wie die UAC in Windows 7 hätte sein können.

In jedem Fall sollte sich Microsoft entscheiden, ob die UAC nun Sicherheits-Feature sein soll oder nicht. Falls nicht, sollten die Entwickler das Icon der Benutzerkontensteuerung überdenken. Das Schild-Symbol impliziert bei den meisten Usern nach wie vor eine Schutzfunktion. Aus marketingtechnischer Sicht sicher nicht verkehrt.

Administrator versus Administratoren

Also, ich hab‘s bis jetzt nicht geschafft, eine klare Meinung darüber auszulassen, wer „mehr“ Rechte hat. (wenn überhaupt). Auch die Microsoft-Hilfe konnte mir da nicht wirklich helfen bei der Frage:

Gibt es einen Unterschied zwischen dem (versteckten) Administrator und der Gruppe der *„Administratoren“*?

Bei der Installation von Vista/Win7 wird ein „echter“ Administrator ohne Kennwort angelegt, aber direkt deaktiviert. Der Account, unter welchem das System installiert wird, ist dagegen schon ein „eingeschränkter“ Administrator.

Normalerweise brauchst Du den versteckten Administrator nie freischalten, da Du ja mit deinem User automatisch der Gruppe der Administratoren angehörst. Und in Windows werden nun mal über die Gruppenzugehörigkeiten die Rechte geregelt.

Alle anderen erstellten Administratoren sind dagegen „eingeschränkte Administratoren“. Deren Accounts haben zwar die vollen Admin-Rechte; werden diese aber benötigt, schaltet sich UAC ein und gibt die „erweiterten“ Rechte für die jeweilige Aktion erst nach Bestätigung frei.

Das versteckte Administratorkonto wird von Windows 7 bei der Installation erstellt und hat nicht mehr Rechte als jeder andere Nutzer der zur Administratoren-Gruppe gehört. Soweit die Theorie.

Dieses versteckte Administrator-Konto ist standardmäßig von den UAC-Meldungen „befreit“ (weil Windows 7 es zur Installation benutzt), d.h. wenn Du mit diesem Konto arbeitest warnt Dich die UAC vor gar nichts. Auch klar.

Warum aber beim Administratorkonto gewisse Sachen deaktiviert werden (...die erwähnte Hardwarebeschleunigung....), und ob es da vor allem bei den Rechten noch Unterschiede gibt. Ich weiß es nicht.

Ich selber ertappe mich aber immer öfters dabei, dass ich unter Windows mit dem versteckten Administrator arbeite. Vielleicht bin ich es so von Linux gewohnt, dass man eine erhöhte Instanz explizit aufrufen und authentifizieren muss. Macht der Gewohnheit.

Manchmal kommt es mir auch vor, dass ich mit dem versteckten Admin mehr Rechte als mit der Gruppe der Administratoren habe. Dieser Eindruck rührt wahrscheinlich daher, weil die UAC dort nicht mehr aufpoppt, was zu erwarten war.

Wer Erfahrungen diesbezüglich hat, kann mich gerne kontaktieren.

Fazit

Die Beherrschung der Rechteverwaltung bedingt einen Lern- und Zeitaufwand, der nicht zu unterschätzen ist. Gilt für Windows und Linux gleichermaßen.

Wer heutzutage mit dem PC arbeitet, kommt zwangsläufig mit der Rechteverwaltung in Berührung. Es geht definitiv kein Weg dran vorbei. Gilt für Linux. In Windows kannst Du die Rechteverwaltung aushebeln. Ob das sinnvoll ist, bezweifle ich aber…

Versuche bei den Berechtigungen so sparsam wie möglich zu sein. Heißt: Lieber nicht so viel erlauben, als vieles verbieten. Lies Dir den Satz noch mal durch. Gut. Anders ausgedrückt: Eher weniger Rechte geben als mehr entziehen… gilt vor allem in Windows. Erspart eventuell viel Kopfweh…

Bei den Berechtigungen in Linux versuche auf der Kommandozeile zu arbeiten: Übersicht und Schnelligkeit stehen einer Einarbeitungszeit positiv gegenüber. Einmal begriffen, kannst Du dann aber das Zeug jahrelang anwenden.

Wenn Du einen UNIX-Guru von 1980 an Deine Kiste läßt, setzt der Dir auch heute noch die Berechtigungen in zwei Minuten richtig!

Das Linux-Rechtesystem funktioniert schon seit Jahrzehnten so…

Wenn Dich die Rechteverwaltung interessiert *(okay wenn‘s vielleicht draußen regnet… aber sonst geh bitte lieber zum Heurigen)* und Du Dich einarbeiten willst. Nimm Dir Zeit. Viel Zeit. Und vor allem Geduld.

Ehrlicherweise muss man sagen, dass die Rechteverwaltung heutzutage ein notwendiges Übel geworden ist, die der eigenen Sicherheit dient. Für viele Windows-User ist das Neuland. Aber das geht vorbei…

Somit neigt sich auch dieser Artikel seinem unerbittlichen Ende zu.

Eines kannst du dir aber sicher sein: Trotz der gewaltigen Ausmaße dieses Artikels *(jäh… endlich hob i des Heftl für mi allanich :-)* behandelte dieser lediglich 10% der Rechteverwaltung; wenn überhaupt. Rechteverwaltung ist ein „Hund“. Das steht einmal fest.

Selbstredend, dass die Zeilen über ein paar Monate hinweg von mir zusammengeschrieben und von Franz Fiala entsprechend aufpoliert wurden.

Nachdem ich so die übliche Portion Nektar um des Chefredakteurs Lefzen gewuchtet *(ergo Honig ums M**l geschmiert)* habe, und Du das Gelesene erst einmal einwirken lassen solltest, ziehe ich mich nun elegant mit einem ausdrucksstarken Punkt am Ende dieses Satzes aus der Affäre .

Man liest sich! Gruß Günter