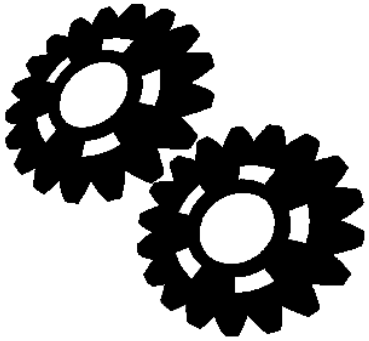


tuck-online
it-systems and networking business

Installationshandbuch

für

Microsoft Windows Server 200x

von Stephen Tuck

Inhaltsverzeichnis

Vorwort

Die hier dargestellten Informationen basieren auf Hintergrundinformationen aus den Büchern „Integrationshadbuch *Microsoft*-Netzwerke 2. Auflage“, „Exchange 2003“ und „Windows 2000 im Netzwerkeinsatz“. Die beschriebenen Installationsroutinen sind keine genau einzuhaltenden Vorgehensweisen, jedoch in der Praxis zu realisieren. Ich bitte um Entschuldigung für etwaige Schreibfehler oder Zeichensetzungsfehler. Da es mir nicht möglich ist zu wissen, ob die von Ihnen gewartete Umgebung zerschossene Betriebssysteme beherbergt, distanziere ich mich hiermit gegenüber jeglicher Haftung durch Installationsprobleme oder Systemausfällen resultierend aus den hier beschriebenen Installationsschritten. Eingetragene Namen werden von mir gekennzeichnet jedoch nicht mit den entsprechenden Symbolen sondern in der *kursiv* Schriebweise. Befehlsauszüge werden in der Schriftart `Courier New` gekennzeichnet.

Heraufstufung zum Domänencontroller (Einrichtung eines Active Directory Services)

Die Heraufstufung zu einem Domänencontroller kann bei einem *Microsoft Windows 2003 Server* durch den Assistenten aufgerufen werden. Es ist aber auch möglich das Programm über einen Befehl aufzurufen. Die hier beschriebene Variante basiert auf dem Befehlsaufruf. Durch einen Mausklick auf `Start` **à** `Ausführen` kann nun ein Befehl ausgeführt werden. Durch die Bestätigung des Befehls `dcpromo.exe` wird der Prozess der Heraufstufung gestartet. Während der Heraufstufung ist es wichtig zu wissen, ob eine neue Gesamtstruktur eingerichtet werden soll, eine untergeordnete Domäne in einem vorhandenen Forest erstellt werden soll, oder vielleicht nur ein zusätzlicher Server für den Domänencontroller als Auslastung. In diesem Beispiel gehe ich davon aus, dass es noch keine Domäne und einen Domänencontroller gibt. Aus diesem Grund wird eine Domäne namens `TESTDOM` sprich `testdom.local` eingerichtet.

Einrichtung eines WINS Servers (Windows Internet Name Service)

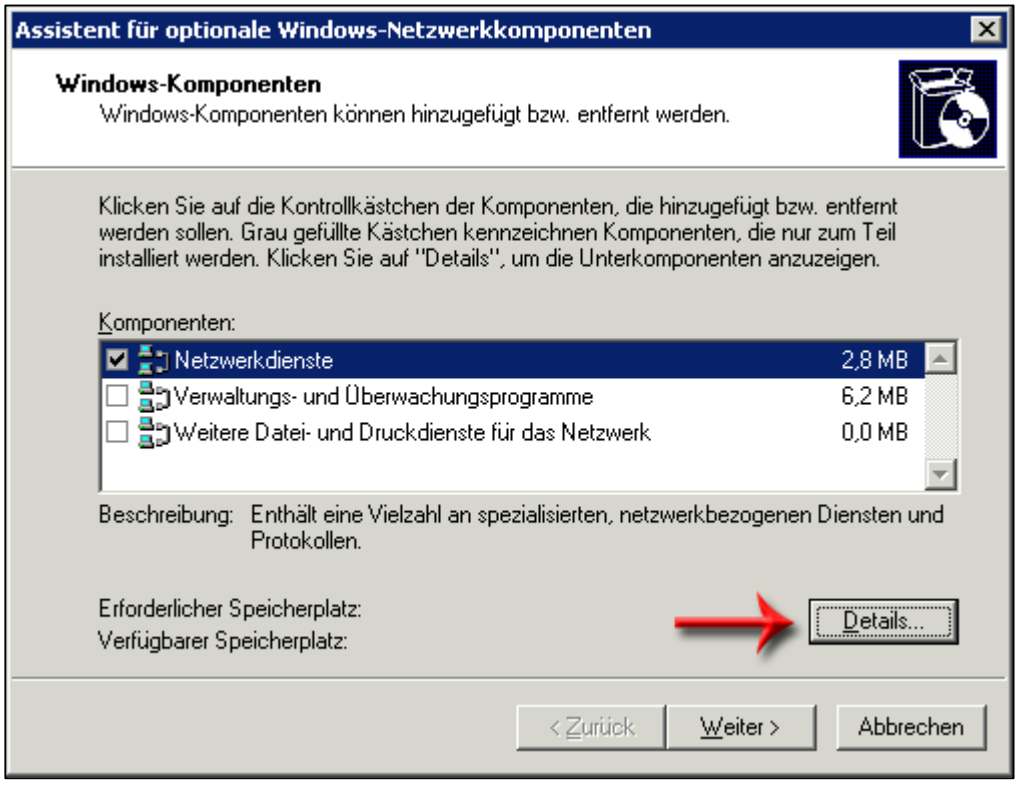
Was tut ein WINS-Server?

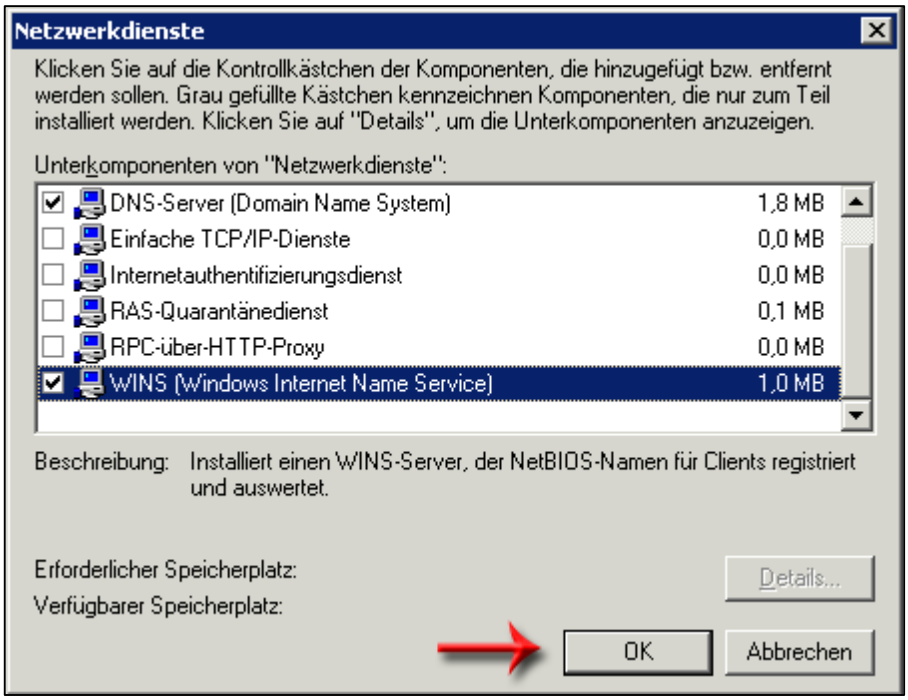
Ein *WINS-Server* wird heutzutage leider immer noch in einem *Microsoft* Netzwerk benötigt, da viele Anfragen übers Netzwerk von *Microsoft* Betriebssystemen auf das *NetBIOS*-Protokoll nicht verzichten können und eine einwandfreie Identifikation bzw. Auflösung der Anfrage zu gewährleisten. Daher ist leider immer noch ein *WINS-Server* neben einem *DNS-Server* notwendig. Der Einsatz eines *WINS-Servers* macht die Nutzung und Verwaltung der Datei `lmhosts` überflüssig und verhindert die netzbelastenden Rundsprüche bei der *NetBIOS*-Namensauflösung. *WINS* ist ein dynamischer Namensdienst, der die Verwaltung in Verbindung mit den *WINS-Clients* selbständig abwickelt und so gut wie keinen Eingriff eines Administrators erfordert. Im Prinzip ist es so, dass sich *WINS-Server* und *WINS-Client* die Arbeit teilen. Wenn der *WINS-Client* gestartet wird, meldet er seinen *NetBIOS*-Computernamen und seine IP-Adresse an den *WINS-Server*. Dieser trägt Name und Adresse in seine Datenbank ein. Will eine Arbeitsstation einen Computernamen aufgelöst haben, nimmt sie Kontakt zum *WINS-Server* auf. Der sieht in seiner Datenbank nach und liefert gegebenenfalls die IP-Adresse zurück. Prinzipiell gilt, dass jeder Name in die *WINS*-Datenbank eingetragen wird, sofern er nicht bereits verwendet wird. In der Regel reicht ein *WINS-Servers* in einem *Microsoft*-Netzwerk aus, jedoch würde bei einem Ausfall die Verarbeitung von Anfragen ins Nirwana laufen und die Performance des Netzwerkes enorm beeinträchtigen. Deshalb empfiehlt sich in großen Netzwerken einen Primary und einen Secondary *WINS-Server* zu installieren. Die Server sollten voneinander unabhängig und redundant ausgelegt sein. Da die beiden *WINS-Server* ihren Datenbestand regelmäßig abgleichen, führt der Ausfall eines Servers nicht zum Stillstand des gesamten Netzwerkes.

Installation und Konfiguration des WINS-Servers

Die Installation und Konfiguration eines *WINS-Servers* ist sehr einfach, da es keinen Konfigurationsaufwand gibt. Durch den Aufruf der `Eigenschaften` in der `Netzwerkumgebung` kann per Mausklick

auf `Erweitert` **à** `Optionale Netzwerkkomponenten` **à** `Netzwerkdienste` „`Details…`“ der *WINS-Server* ausgewählt werden. Dies geschieht durch das setzen eines Häkchens.





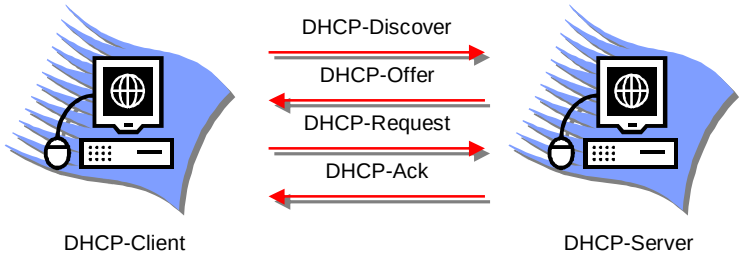
Einrichtung eines DHCP Servers (Dynamic Host Configuration Protocol)

Um ein Netzwerk per *TCP/IP* aufzubauen ist es notwendig jede einzelne Station zu konfigurieren. Bei größeren Netzwerken ist sehr viel Planung- und Arbeitszeit notwendig. Um dem zu entgehen, wird *DHCP* für die vollautomatische Konfiguration von *TCP/IP* verwendet. Doch nicht nur das, *DHCP* ist in

der Lage IP-Adressen zu verwalten und dynamisch zu verteilen. So muss einer Station nicht mehr unbedingt eine feste IP-Adresse zugewiesen werden. Für ein *TCP/IP*-Netzwerk müssen folgende Einstellungen bei jeder Station vorgenommen werden:

- Vergabe einer eindeutigen IP-Adresse
- Zuweisen einer Subnetzmaske (Subnetmask)
- Zuweisen des Default- bzw. Standard-Gateways

Funktionsweise von DHCP



DHCP ist eine Client-Server-Architektur. Der *DHCP-Server* verfügt über einen Pool von IP-Adressen, die er den *DHCP-Clients* frei zuteilen kann. Bei größeren Netzen muss der *DHCP-Server* zudem wissen, welche Subnetze und Standard-Gateway es gibt. Wird eine Station gestartet und ist dort ein *DHCP-Client* aktiviert, wird ein in seiner Funktion eingeschränkter Modus des *TCP/IP*-Stacks gefahren. Dieser hat keine gültige IP-Adresse, keine Subnetzmaske und kein Standard-Gateway. Das einzige, was der Client machen kann, ist IP-Broadcasts zu verschicken. Der *DHCP-Client* verschickt ein UDP-Paket mit der Ziel-Adresse 255.255.255.255 und der Quell-Adresse 0.0.0.0. Dieser Broadcast dient als Adressanforderung an alle verfügbaren *DHCP-Server*. Das UDP-Paket enthält die Hardware-Adresse (MAC-Adresse) der Station. Jeder angesprochene *DHCP-Server* schickt daraufhin ein UDP-Paket mit folgenden Daten zurück:

- MAC-Adresse des Clients
- mögliche IP-Adresse
- Laufzeit der IP-Adresse
- Subnetzmaske
- IP-Adresse des *DHCP-Servers* / Server-ID

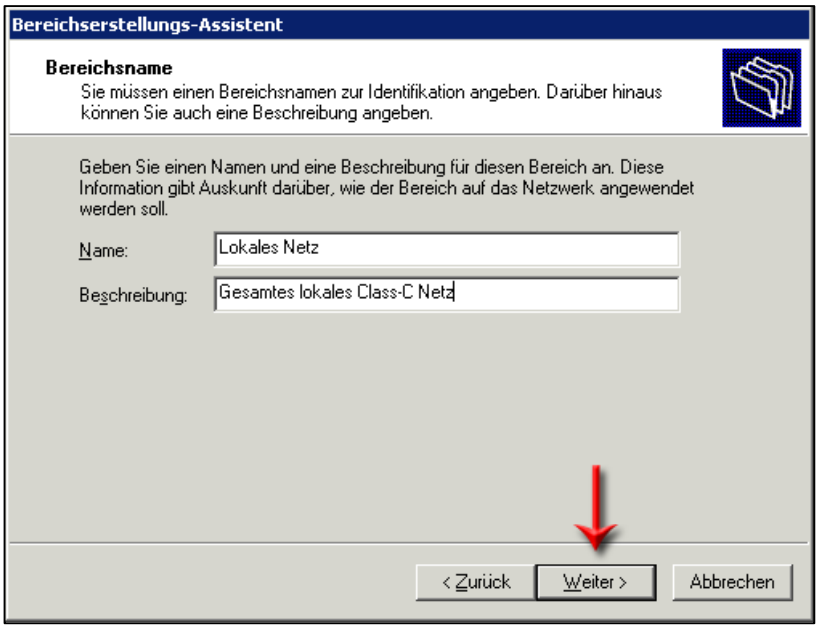
Aus der Auswahl von eventuell mehreren *DHCP-Servern* sucht sich der *DHCP-Client* eine IP-Adresse heraus. Daraufhin verschickt es eine positive Meldung an den betreffenden *DHCP-Server*. Alle anderen Server erhalten die Meldung ebenso und gehen von der Annahme der IP-Adresse zugunsten eines anderen Servers aus. Anschließend muss die Vergabe der IP-Adresse vom *DHCP-Server* bestätigt werden. Sobald der *DHCP-Client* die Bestätigung hat, speichert er die Daten lokal ab. Abschließend wird der *TCP/IP*-Stack vollständig gestartet. Doch nicht nur die Daten zum *TCP/IP*-Netzwerk kann *DHCP* an den Client vergeben. Sofern der *DHCP-Client* weitere Angaben auswerten kann, übermittelt der *DHCP-Server* weitere Optionen:

- Time Server
- Name Server
- *Domain Name Server*
- *WINS-Server*
- Domain Name
- Default IP TTL
- Broadcast Address
- *SMTP Server*
- *POP3 Server*

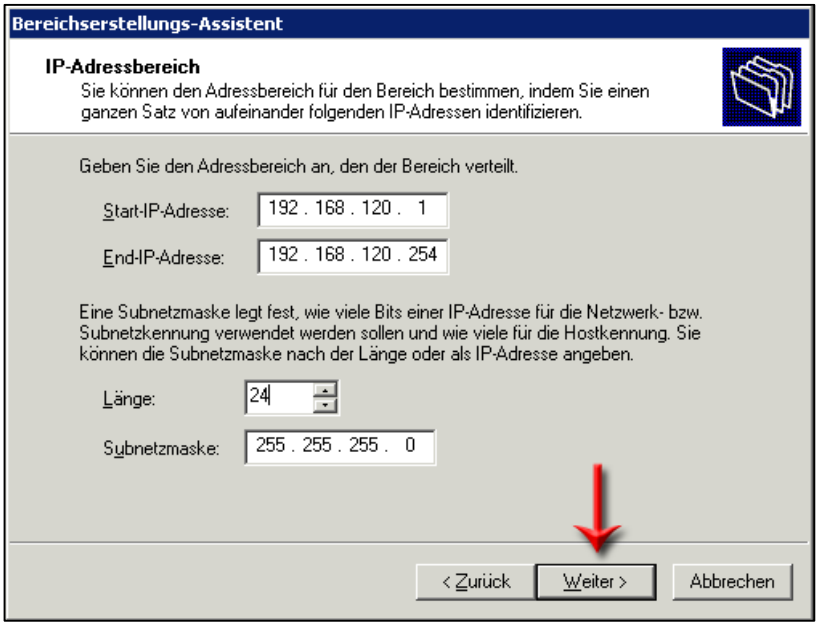
Installation und Konfiguration des DHCP-Servers

Die Installationsroutine ist fast genauso wie beim *WINS-Server*. Es wird bei der Auswahl des zu installierenden Dienstes nicht der *WINS-Sever* sondern der *DHCP-Server* ausgewählt.

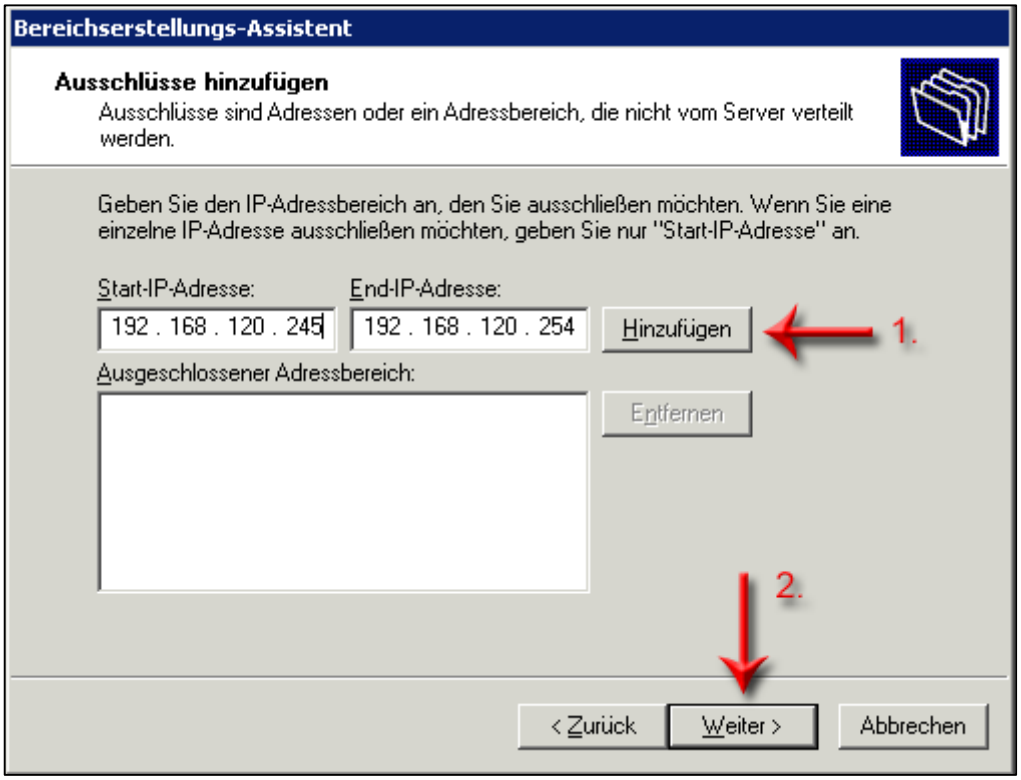
Nach erfolgter Installation kann das Snap-In `DHCP` unter `Start` **à** `Einstellungen` **à** `Systemsteuerung` **à** `Verwaltung` **à** `DHCP` geöffnet werden. Mit einem Rechtsklick auf das *DHCP-Server* Symbol mit einem roten Pfeil nach unten und der Auswahl `Neuer Bereich` im Kontextmenü kann der Server konfiguriert werden. Ein Assistent startet und kann mit `Weiter` bestätigt werden. Da ein Neuer Bereich eingerichtet wird muss ein Name für diesen hinterlegt werden. Optional können weitere Informationen unter Beschreibung abgelegt werden.



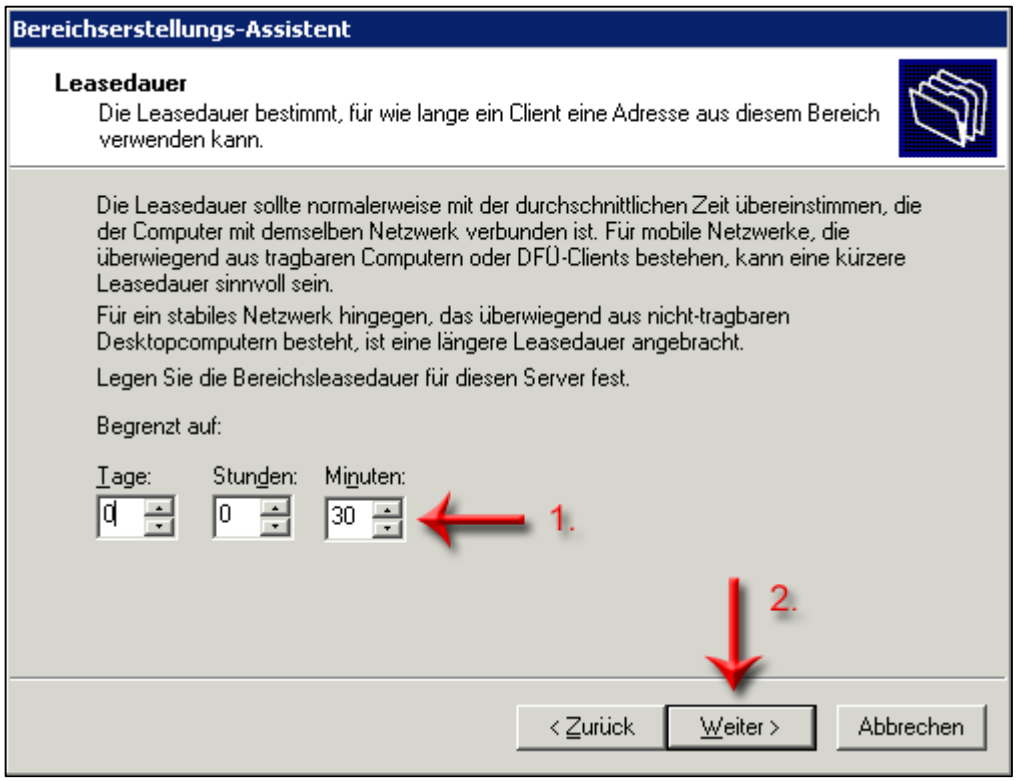
Nach der Bestätigung mit `Weiter` kann nun der IP-Adressbereich für die dynamisch zugewiesenen IP-Adressen definiert werden. Hierfür muss eine Start- und End-IP-Adresse zugewiesen werden (z.B. Class C-Netz 192.168.x.1 bis 192.168.x. 254). Damit wären vorerst 254 Hostadressen für den *DHCP-Server* reserviert. Zusätzlich kann auch schon die Subnetzmaske angegeben werden. Für das gesamte Class-C Netz verwendet man die Länge 24 oder die IP-Adresse `255.255.255.0`.



Da aber in diesem Bereich garantiert die IP-Adresse des Servers enthalten ist, müssen bestimmte Adressen von der dynamischen Vergabe ausgeschlossen werden. Es empfiehlt sich den *Router* und die Netzwerkdrucker sowie *Printserver* mit festen IP-Adressen zu versehen. In diesem Beispiel habe ich 10 IP-Adressen ausgeschlossen (192.168.x.245 – 192.168.x.254).



Daraufhin ist es möglich die Lease-Time einzustellen. Theoretisch könnte man den Wert so hoch stellen, damit die vergebenen IP-Adressen fast schon wie statische wirken. In diesem Beispiel stelle ich 30 Minuten als Ablaufzeit ein.



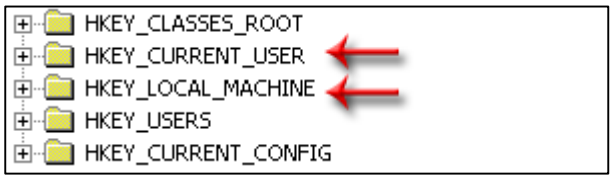
Nun können bei Bedarf auch noch zusätzliche Informationen an die *DHCP-Clients* weitergegeben werden (z.B. *Router*, *DNS*, *WINS*). Aus diesem Grund bestätigt man das Fenster mit `Ja, diese Optionen jetzt konfigurieren`. Es wird nach der IP-Adresse des *Routers* in dem Netzwerk gefragt. Dieser sollte ja wie schon zuvor erwähnt in dem ausgeschlossenen Bereich liegen. Danach kann der *DNS*-Suffix bestimmt werden und die IP-Adressen der vorhandenen *DNS-Server* in dem lokalen Netzwerk sowie die IP-Adressen der *ISP DNS-Server*. Es ist nur wichtig, dass die lokalen Server vorrangig vor den *DNS-Servern* des Providers angefragt werden. Ansonsten würde immer eine Einwahl ins Internet geschehen, wenn nach einem lokalen Namen oder IP-Adresse gefragt wird. Jetzt ist es möglich nach einem Mausklick auf `Weiter` die IP-Adresse des *WINS-Servers* anzugeben. Abschließend kann der Bereich aktiviert werden, oder aber auch erst offline betrieben werden, bis es nötig ist diesen zu aktivieren. Wichtig ist jetzt noch die Autorisierung des *DHCP-Servers*, da ansonsten der Server nicht aktiviert ist. Durch einen Rechtsklick auf das *DHCP-Server* Symbol (roter Pfeil nach unter) wird das Kontextmenü aufgerufen. Hier kann der Punkt `Autorisieren` angewählt werden. Der rote Pfeil wechselt in einen grünen Pfeil der senkrecht nach oben ausgerichtet ist.

Verwendung von Gruppenrichtlinien

Was sind Gruppenrichtlinien?

Gruppenrichtlinien sind Sammlungen von Benutzer- und Computerkonfigurationseinstellungen, die mit Computer, Standort, Domänen oder Organisationseinheiten (Organisation Unit) verknüpft werden können, um das Verhalten des Benutzerdesktops zu steuern und darüber hinaus Dinge wie Sicherheitseinstellungen, Anmelde- und Abmeldescripte, Skripte für den Start und das Herunterfahren eines zu definieren oder z.B. Ordnerumleitungen festzulegen. Mit Gruppenrichtlinien (Group Policy Object) kann das Verhalten des Betriebssystems bestimmt und dessen Optionen können eingeschränkt werden. Es gibt aber auch Gruppenrichtlinien, mit denen das Verhalten und die Optionen von Anwendungen wie z.B. *Microsoft Office* von zentraler Stelle aus gesteuert werden können.

Genauer betrachtet sind Gruppenrichtlinien Registrierungsschlüssel, die in regelmäßigen Abständen angewandt werden. Hier ist zu beachten, dass nur die Registrierungsgruppen `HKEY_CURRENT_USER` (Benutzerkonfigurationen) und `HKEY_LOCAL_MACHINE` (Computerkonfigurationen) diesen Part übernehmen. Im Klartext heißt das, dass jede im *Active Directory* hinterlegte Gruppenrichtlinie ein ganz simpler Registrierungsschlüssel mit einem Wert ist.



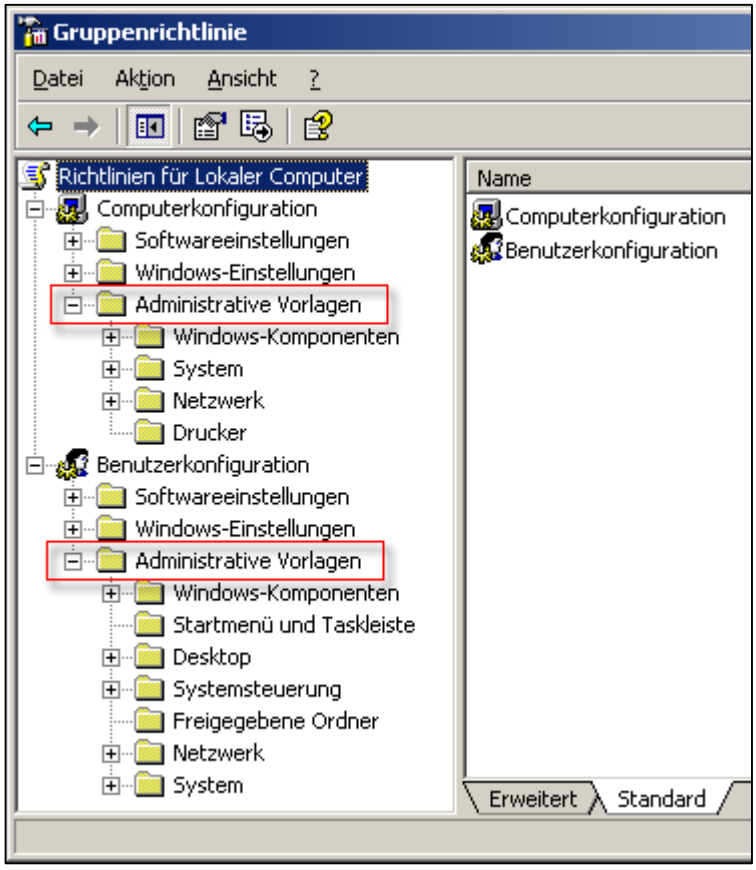
Was sind Gruppenrichtlinienverknüpfungen?

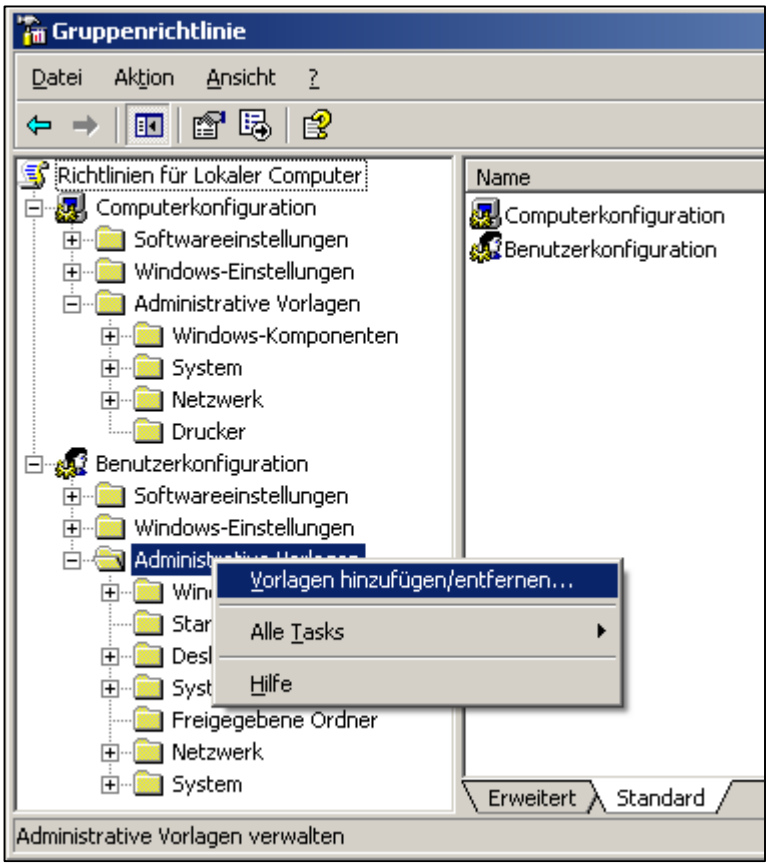
Bestimmte Gruppenrichtlinien sind abteilungsspezifisch, andere sollen für alle Mitarbeiter der Organisation gelten. Die Richtlinien, die für alle Mitarbeiter gelten sollen, definiert man nur einmal in einer Gruppenrichtlinie, die für diesen Zweck in einer Organisationseinheit erzeugt wird. Anschließend verknüpft man diese Gruppenrichtlinie mit allen Abteilungen (Organisationseinheiten: z.B. Vertrieb, Verwaltung, Einkauf, Produktion, etc.). Falls man später eine organisationsübergreifende Richtlinie hinzufügen oder anders definieren möchte, nimmt man diese Änderung komfortabel an einer zentralen Stelle und nicht über jede einzelne Organisationseinheit (Abteilung). Die Anzahl der Gruppenrichtlinien und möglichen Fehlerquellen bleibt überschaubar.

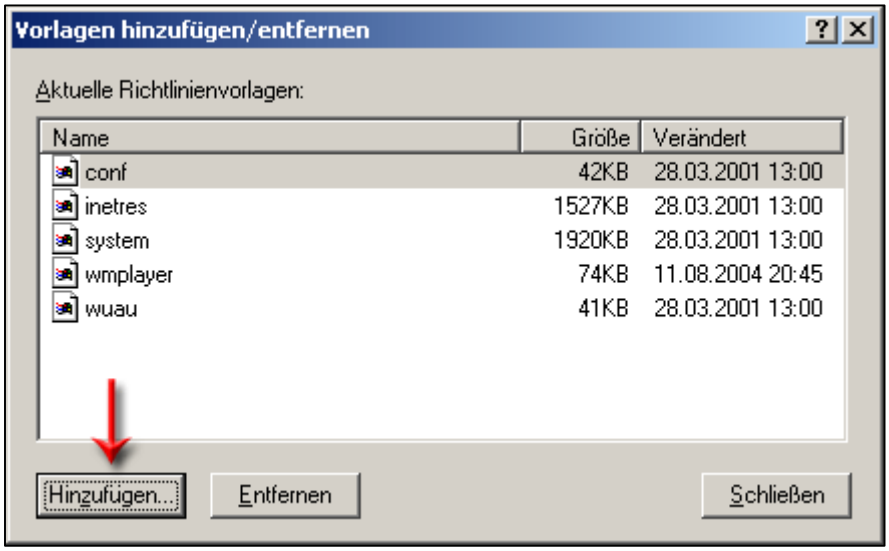
Was sind Gruppenrichtlinienvorlagen?

Gruppenrichtlinienvorlagendateien haben die Dateierweiterung `*.adm`. Diese Dateien stellen die einzelnen Informationen für den Ordner *Administrative Vorlagen* da. Um dies einfacher und deutlicher zu beschrieben habe ich einige Screenshots gemacht. Als erstes wird per Rechtsklick auf eine Organisa-

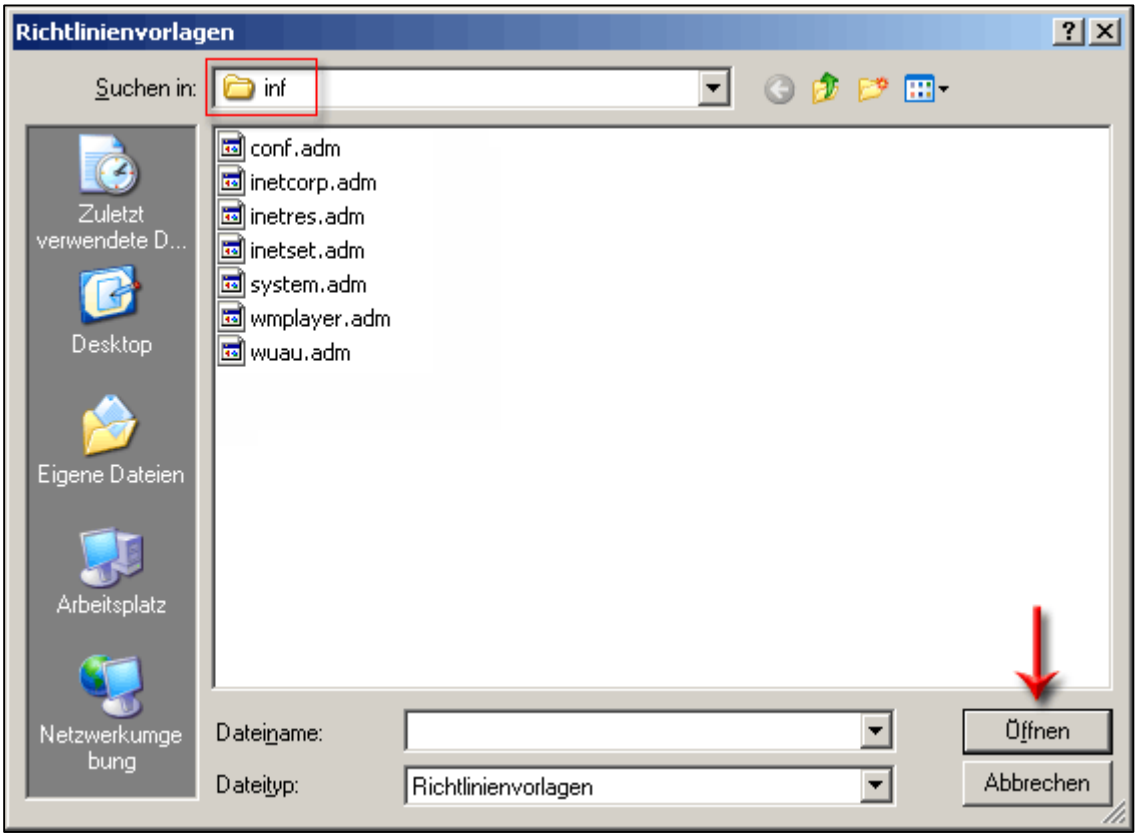
tionseinheit das `Gruppenrichtlinien` Snap-In über die `Benutzer- und Computer` Verwaltung geöffnet. Danach wählt man mit der rechten Maustaste im Kontextmenü bei `Administrative Vorlagen`, `Vorlagen hinzufügen/entfernen...` aus.







Die erscheinenden `*.adm` Dateien können nun per Mausklick auf `Hinzufügen` ausgewählt werden. Wie man erkennen kann wird standardisiert das `inf` Verzeichnis des Betriebsystems ausgewählt. Im *Windows-Explorer* kann man ebenfalls die `*.adm` Dateien sehen, wenn man in das Systemverzeichnis `inf` im Windows Ordner wechselt (C:\Windows\inf).



Anwendungen wie *Microsoft Office* bieten für den Systemadministrator die Möglichkeit mit dem kostenlosen *Office Resource Kit* `*.adm` an. Diese werden automatisch bei der Installation in den Konfigurationsdateien Ordner kopiert. Leider muss man diese nachträglich über den oben beschriebenen Weg hinzufügen. Dadurch ist es ebenfalls möglich neben dem Betriebssystem auch noch Anwendungen anzupassen.

Softwareverteilung

Verwendete Methode der Softwareverteilung

Unter Softwareverteilung versteht man die Verteilung von Anwendungen übers Netzwerk. Dadurch ist es möglich zeitsparender und mit weniger Aufwand Software auf Arbeitsstationen zu installieren. Besonders einfach wird es, wenn der Administrator auch noch das Betriebssystem aus dem Netzwerk installieren lässt uns alle Konfigurationen automatisiert wurden. Eine häufig verwendete Möglichkeit der Softwareverteilung von Applikationen ist die Installation per Anmeldescript und der Befehlsoption die Anwendung im Hintergrund und ohne Benutzereingriff zu installieren. Jedoch ist bei dieser Lösung der größte Nachteil, dass nur sehr umständlich die Installation angepasst werden kann während der Installationsphase. Meist ist es nicht möglich, oder der Administrator muss sich per Fernwartung auf die Arbeitsstation verbinden und alle Einstellungen manuell durchführen. Die einfachste Lösung ist hier der Einsatz von *Microsoft Windows Installer* Dateien (*MSI*). Diese können von einer Freigabe des Servers per Gruppenrichtlinie auf die Arbeitsstation Anwendungen verteilen ohne viel Schreib- und Zeitaufwand. Sicherlich ist es mindestens genauso einfach Software mit professionellen Lösungen wie *Netinstall* übers Netzwerk zu installieren, aber ich möchte in diesem Artikel auf eine kostengünstige und schnelle Lösung eingehen. Viele Softwarehersteller liefern schon eine *MSI*-Datei zur Softwareverteilung Ihrer Applikation mit. In diesem Fall ist der Aufwand wirklich nur noch minimal, jedoch ist es hier meist auch erwünscht eine Anpassungsdatei für die Installation zu erstellen. Leider gibt es zu diesem Thema nicht so viele Lösungsansätze von Drittherstellern bzw. die Softwarehersteller liefern eine Anpassungssoftware mit wie z.B. *Microsoft*, *McAfee*, *Adobe* etc.

Dieser Artikel umfasst ausschließlich die zugewiesene Verteilmethode eines *Microsoft Windows 2003 Servers* auf Computerebene. Es gibt ebenfalls die Möglichkeit Software auf einer Freigabe für die Benutzerinstallation zu verwalten, aber die meist gewünschte Installationsmethode ist die zugewiesene computerbasierte Softwareverteilung.

MSI-Datei Erstellung

Die Firma *OnDemand* bietet mit der Software *WinInstall LE 2003* sogar eine Lösung an, mit der man Anwendungen in *MSI*-Dateien verpacken kann. Dies ist beispielsweise notwendigen, wenn der Softwarehersteller einer Applikationen keine *MSI*-Version beigelegt hat. Diese abgespeckte Version war umsonst erhältlich von *OnDemand*. Mittlerweile kostet diese Version ca. EUR 50. Unter der Downloadrubrik habe ich die vorherige, kostenlose Version freigegeben. Die Vorbereitung einer *MSI*-Datei ist etwas aufwändiger. Vorab muss eine Arbeitsstation mit dem gewünschten Betriebssystem versehen werden, da dort die zu verteilende Anwendung installiert werden muss. Hat man mehrere Anwendungen die auf diese Weise vorbereitet werden müssen, empfiehlt es sich von dem frisch installierten Betriebssystem ein Abbild zu erstellen. Hierzu können Anwendungen wie *Symantec Norton Ghost* oder *PowerQuest DriverImage* dienen. Das noch jungfräuliche Windows wird nun mit allen nötigen Updates versehen und kann dann als Installationsreferenz dienen.

Nach der Installation des *WinInstall LE 2003* kann ein Abbild des Systems erstellet werden. Dies ist notwendig, weil die Software nach dem Vergleichsprinzip arbeitet. Alle Einstellungen und Dateien sowie die Registrierung werden protokolliert. Danach kann ganz gewöhnlich die gewünschte Anwendung installiert werden. Abschließend wird noch mal ein Abbild des Systems per *WinInstall LE 2003* gemacht. Beide Abbilder werden miteinander verglichen und alle Änderungen werden in der *MSI*-Datei festgehalten. Nun ist eine *MSI* basierende Version der Anwendung erstellt worden.

Lieder ist die hier erwähnte Erstellung von Softwarepaketen auf *MSI* Basis kein Freifahrtsschein, da viele Applikationen diese „Umwandlung“ nicht verarbeitet können und es bei einer Installation auf einer Arbeitsstation zu Fehlern kommen kann sowie beim Aufruf der Software durch den Benutzer.

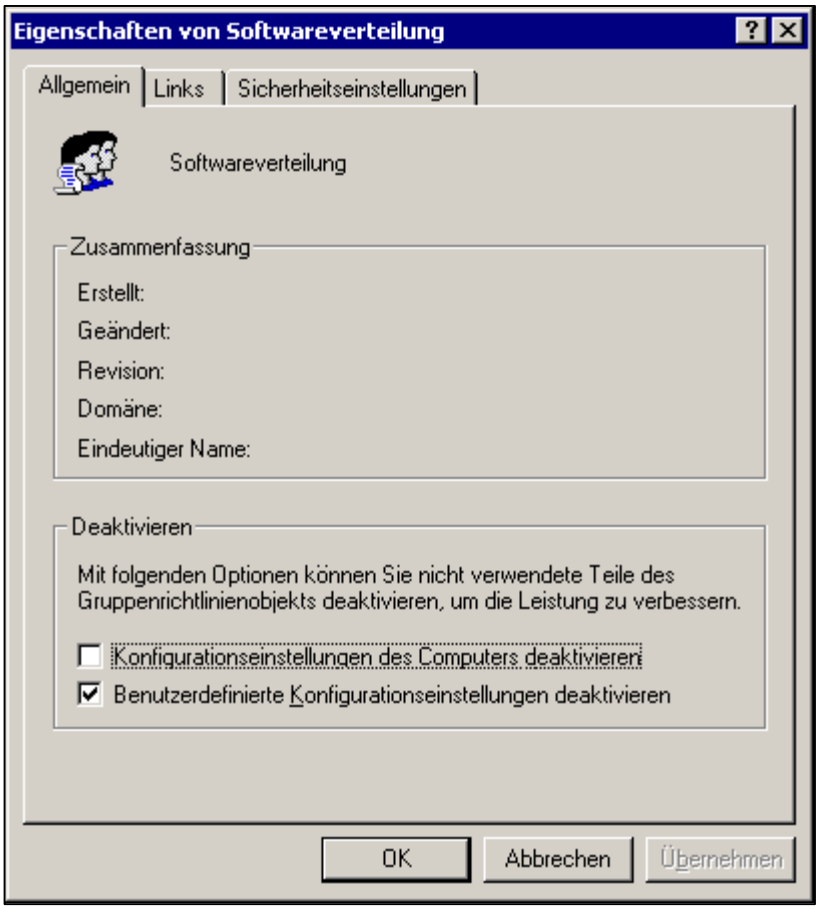
Anpassungsdateien für die Softwareverteilung

Wie schon weiter oben erwähnt gibt *Microsoft* die Möglichkeit z.B. das Officepaket während der Installation anzupassen. Das so genannte *Office Resource Kit* ist kostenlos von der *Microsoft* Unternehmenswebpräsenz herunterladbar. Ebenso sind die Gruppenrichtlinien Dateien (`*.adm`) in dem *Resource Kit* enthalten und können per Kopiervorgang in den Systemkonfigurations-Ordner des Servers kopiert werden.

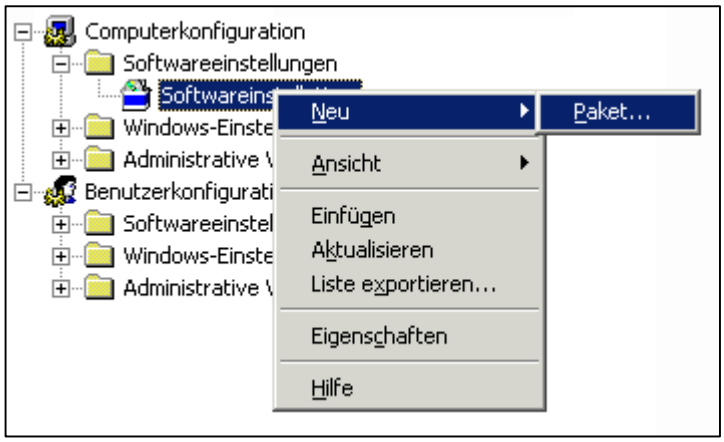
Die Anpassungsdatei (*Microsoft Transformation File*) wird per *Custom Installation Wizard* erstellt. Es ist möglich die Softwarepakete (*MS Access*, *Outlook*, *Word* und *Excel*) anzupassen. Ganz besonders *Microsoft Outlook* kann so angepasst werden, dass bei einem Einsatz des *Exchange Servers* nach der Installation und der ersten Verbindung mit dem Server alle nötigen Konfigurationsschritte automatisiert werden. Somit muss der Anwender nichts mehr konfigurieren, außer in seinem Postfach alltägliche Dinge zu überwachen. Außerdem ist es möglich viele Einstellungen im Vorwege anzupassen z.B. die Ansicht der Menüs in Office (Vollständige Menüs anzeigen). Da diese Einstellungen aber nur die Vorgabe sind und nachträglich vom Anwender jeder Zeit änderbar sind, empfiehlt es sich die Gruppenrichtlinien zu verwalten. Erst dann sind nicht erwünschte Änderungen vom Benutzer ausgeschlossen.

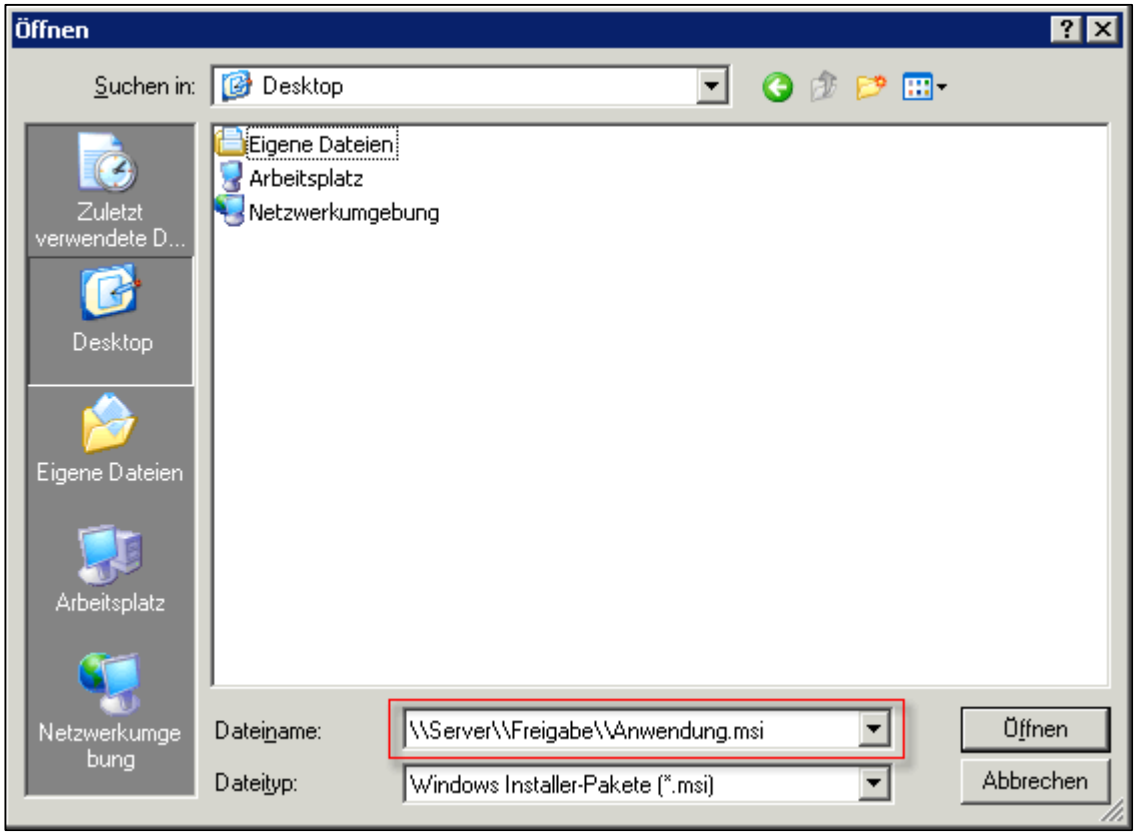
Praxis Beispiel einer computerbasierten Softwareverteilung

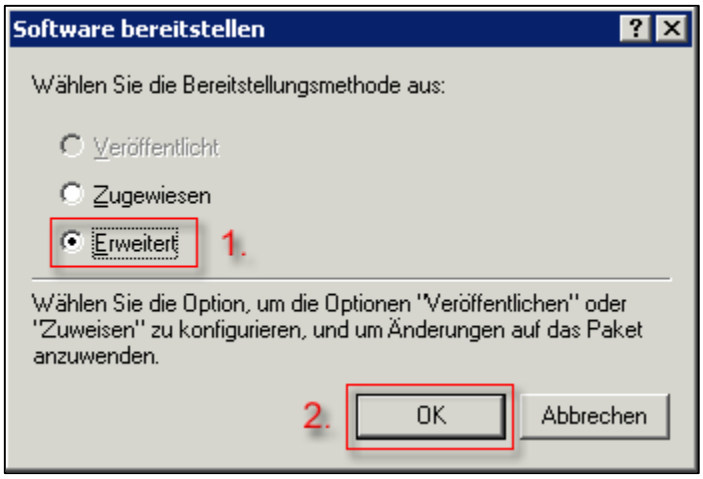
Als erstes wird eine Gruppenrichtlinie angelegt in dem man auf die soeben erstelle Organisationseinheit das Kontextmenü aufruft und dort die `Eigenschaften` aufruft. Mit eine Mausklick auf `Neu` unter der Reiterkarte `Gruppenrichtlinie` kann man eine neue Gruppenrichtlinie anlegen. Diese sollte erst mal in den Eigenschaften bearbeitet werden. Unter `Eigenschaften` **à** Reiterkarte `Allgemein` **à** Unterpunkt `Deaktivierung` wird der Hacken bei `Benutzerdefinierte Konfigurationseinstellungen deaktivieren` gesetzt. Das ist vorteilhaft für die Abarbeitung der Richtlinien bei der Anmeldung, da die Richtlinie ausschließlich Computerspezifische Konfigurationen beherbergt.

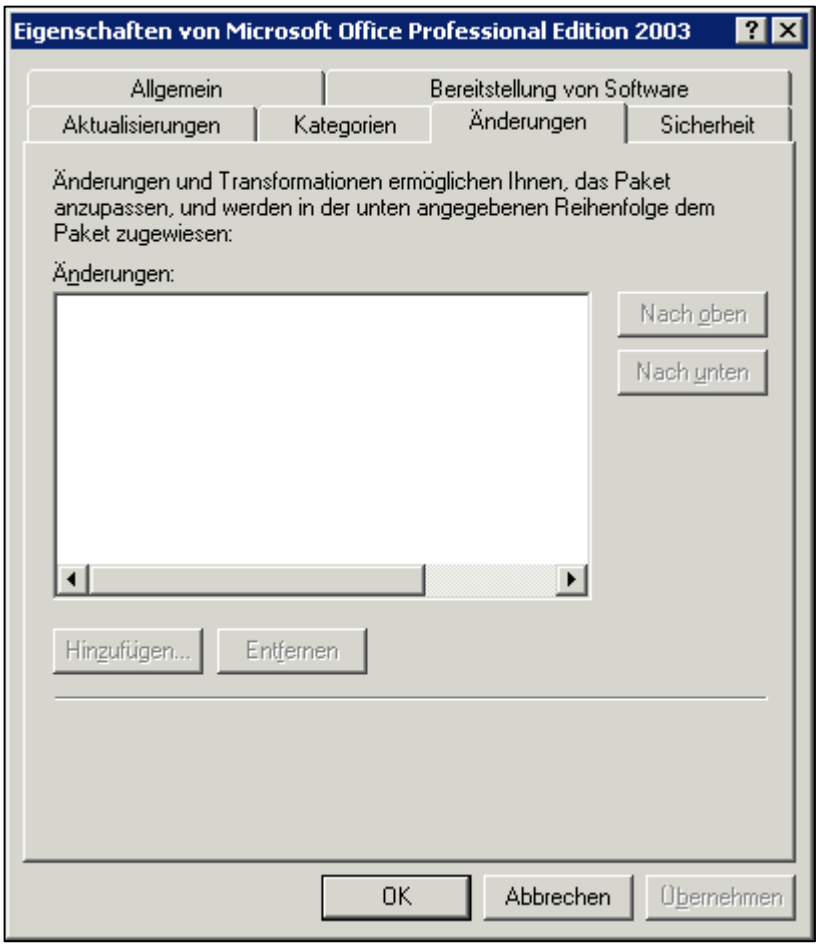


Der reguläre Doppelklick auf die Richtlinie öffnet das `Gruppenrichtlinien` Snap-In. Das Softwarepaket aus der Freigabe wird im Kontextmenü von `Computerkonfiguration` **à** `Softwareeinstelllungen` **à** `Softwareinstallation` **à** `Neu` **à** `Paket…` aufgerufen. Jetzt ist es wichtig die Freigabe anzugeben (UNC-Pfad) ansonsten wird es zu Konflikten bei der Verteilung kommen (lokale Pfade bereiten Probleme). Das Fenster `Software bereitstellen` sollte mit `Erweitert` bestätigt werden, weil nur in diesem Modus das Einpflegen einer *MST*-Datei unter der Reiterkarte `Änderung` gestattet ist.









Einrichten eines DNS Servers (Domain Name Services) mit Active Directory Integrität

Was hinter DNS steckt

Der *Domain Name Service* (*DNS*) ist ein verteiltes, hierarchisches System zur Auflösung von Computernamen in IP-Adressen und umgekehrt. *DNS* kennt keine zentrale Datenbank. Stattdessen sind die Informationen über viele tausend Nameserver (*DNS-Server*) verteilt. Die *DNS*-Datenbank ist eine in Zonen aufgeteilte baumförmige Struktur. Sie beginnt im Root-Verzeichnis. Computernamen, die mit

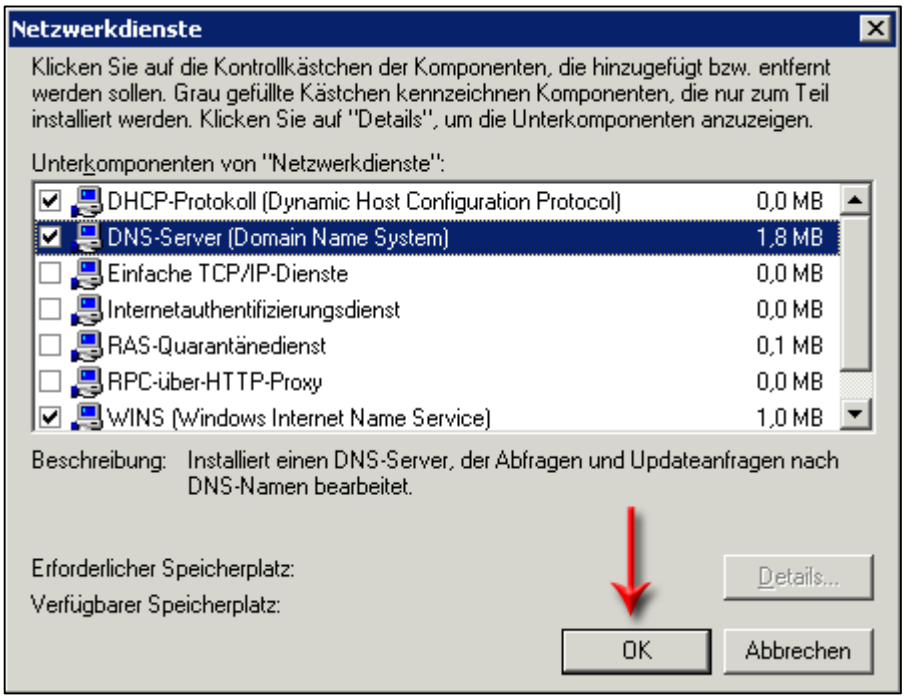
DNS in IP-Adressen aufgelöst werden nennen sich Domain-Namen und haben eine bestimmte Struktur. Sie wird als Uniform Resource Locator (URL), zu Deutsch "einheitliche Angabeform für Ressourcen, bezeichnet. Die für *DNS* verwendeten URLs bestehen aus drei oder mehr Teilen.

Computername (Host oder Dienst)	Second-Level-Domain (SLD)	Top-Level-Domain (TLD)
ftp.	testdom.	de
www.	testdom.	de

Konfiguration und Installation eines DNS-Servers

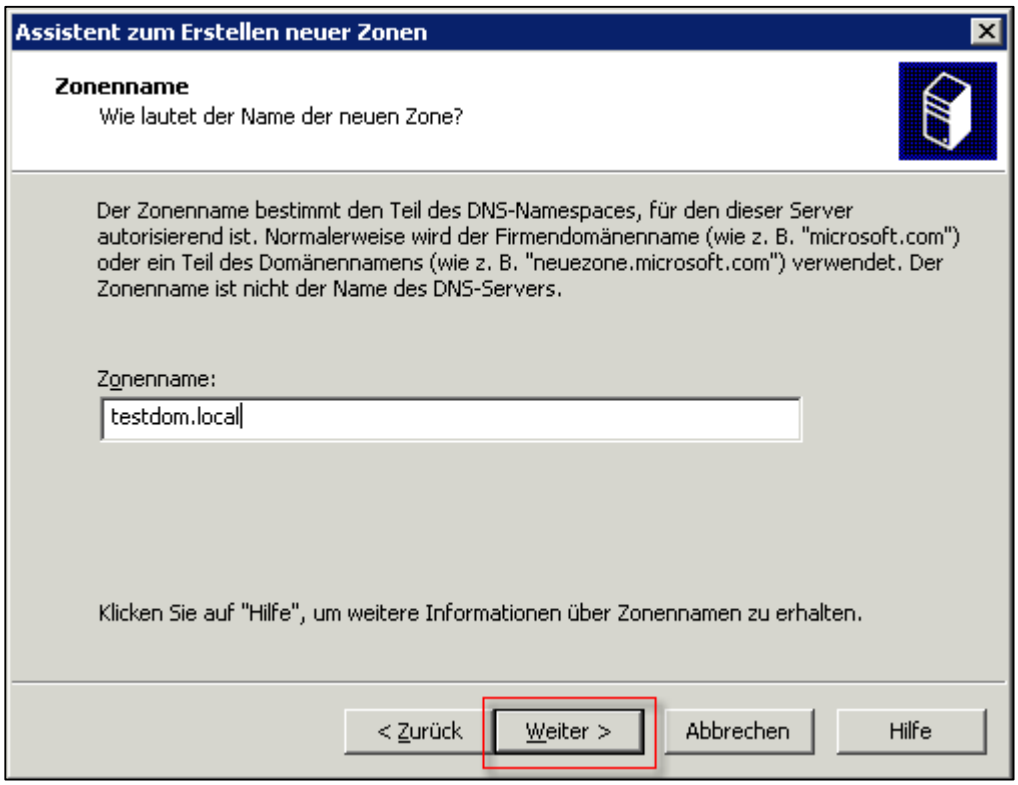
Um eine *DNS-Server* unter einem Windows Server zu installieren, dient meist automatisch die Heraufstufung zum Domänencontroller. Bei der Absetzung des Befehls `dcpromo.exe` wird ein Server zum Domänencontroller heraufgestuft. In diesem Prozess wird gefragt, ob man nicht gleich einen *DNS-Server* mitinstallieren will und dieser dann automatisch konfiguriert werden soll. Diese Möglichkeit sollte man in Erwägung ziehen. Nach dem Abschluss der Hochstufung und einem Neustart kann das Snap-In DNS über `Start` **à** `Einstellungen` **à** `Systemsteuerung` **à** `Verwaltung` **à** `DNS` erreicht werden. Es ist sofort erkennbar, dass nur die *Forward-Lookupzone* konfiguriert wurde. Das hat zur Folge, dass nur die Namen in IP-Adressen aufgelöst werden könne, jedoch nicht umgekehrt. Deshalb ist es notwenig die *Reverse-Lookupzone* manuell nach zu konfigurieren.

Da man aber nicht jeden Tag einen *DNS* im Zusammenhang mit einem *Active Directory* installiert, erwähne ich hier die manuelle Installation. Als erstes wählt man mit einem Rechtsklick im Kontextmenü der `Netzwerkumgebung` die `Eigenschaften` aus. Unter `Erweitert` wählt man `Optionale Netzwerkkomponenten` aus. In der nun erscheinenden Liste wird der Hacken vor `DNS-Server (Domain Name System)` gesetzt und mit `OK` bestätigt.



Nach dem erfolgreichen Kopiervorgang, kann man den *DNS-Server* unter der oben genanten Programmgruppe erreichen. Mit einem Rechtsklick auf `Forward-Lookupzone` erscheint das Kontextmenü. In diesem wählt man den obersten Punkt aus `Neue Zone...`. Ein Assistent wird gestartet und das erste Fenster kann mit `Weiter` bestätigt werden. Bei einem *Microsoft Windows 2003 Server* wählt man den Punkt `Primäre Zone` aus und setzten den Hacken bei `Die Zone in Active Directory speichern (nur wenn der DNS-Server als Domänencontroller eingerichtet ist)`. Danach wird bei Zonendaten replizieren `Auf allen DNS-Servern in der Active Directory-Domäne „testdom.local“` ausgewählt und mit `Weiter` bestätigt. Nun wird nach dem

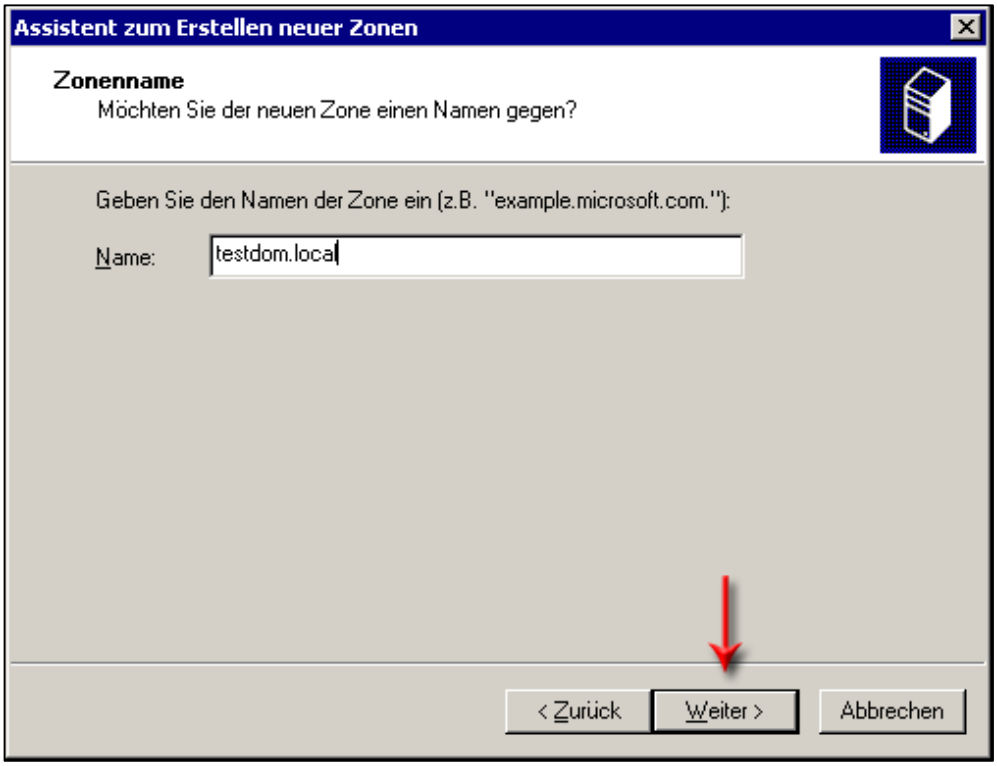
Zonennamen gefragt. An dieser Stelle sollte der Domänenname angegeben werden (Second-Level-Domain.Top-Level-Domain = *testdom.local*).



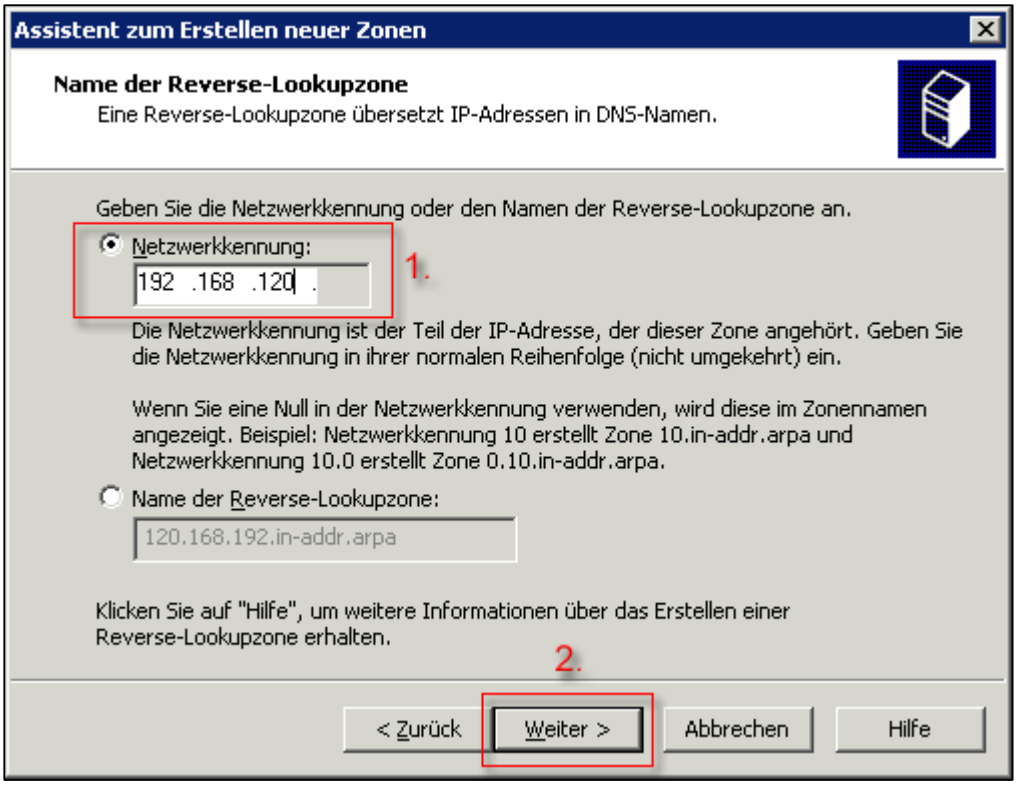
Die Option `Nur sichere dynamische Updates zulassen (für Active Directory empfohlen)` mit `Weiter` bestätigen. Der Assistent kann dann mit einem Mausklick auf `Fertig stellen` beendet werden.

Die *Reverse-Lookupzone* wird mit einem Rechtsklick auf `Reverse-Lookupzone` und der Auswahl im Kontextmenü `Neue Zone...` eingerichtet. Der Assistenten geleitet einen durch die Konfigurationsschritte. Das erste Fenster kann mit `Weiter` gleich bestätigt werden. Beim `Zonentyp` wird wieder die `Primäre Zone` ausgewählt und ein Hacken bei `Die Zone in Active Directory speichern (nur wenn der DNS-Server als Domänencontroller eingerichtet ist)` setzen. Der `Zonenreplikationsbereich` wird wie oben übernommen. Nun ist es notwenig die `Netzwerkkennung` anzugeben. Das dynamische Update Verhalten wird wie oben beschrieben übernommen und das letzte Fenster des Assistenten wird mit `Fertig stellen` beendet. Jetzt ist es dem *DNS-Server* möglich IP-Adressen in Namen aufzulösen.

Bei einem *Microsoft Windows 2000 Server* wählt man beim *Forward-Lookupzonen* Assistenten unter `Zonentypen` nicht `Primär (Standard)` sondern `Active Directory-integriert` aus. Nach der Bestätigung mit `Weiter` wird der Zonenname abgefragt (SLD.TLD = testdom.local).

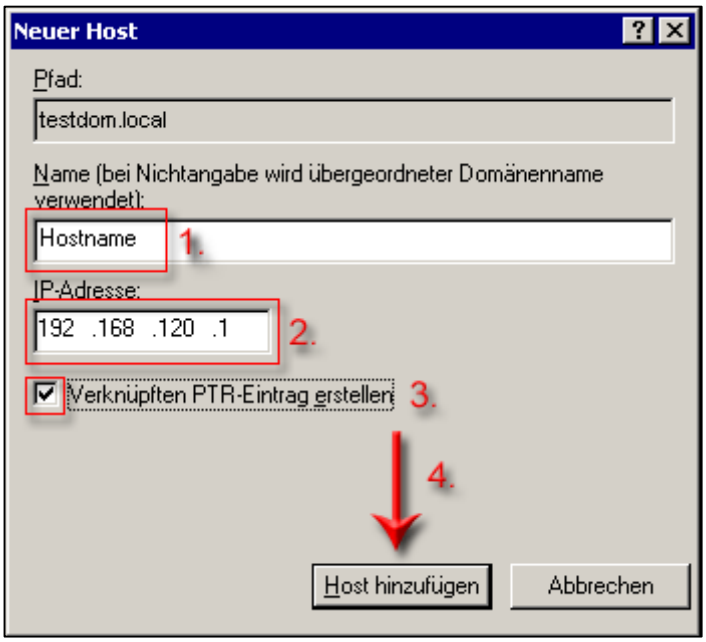


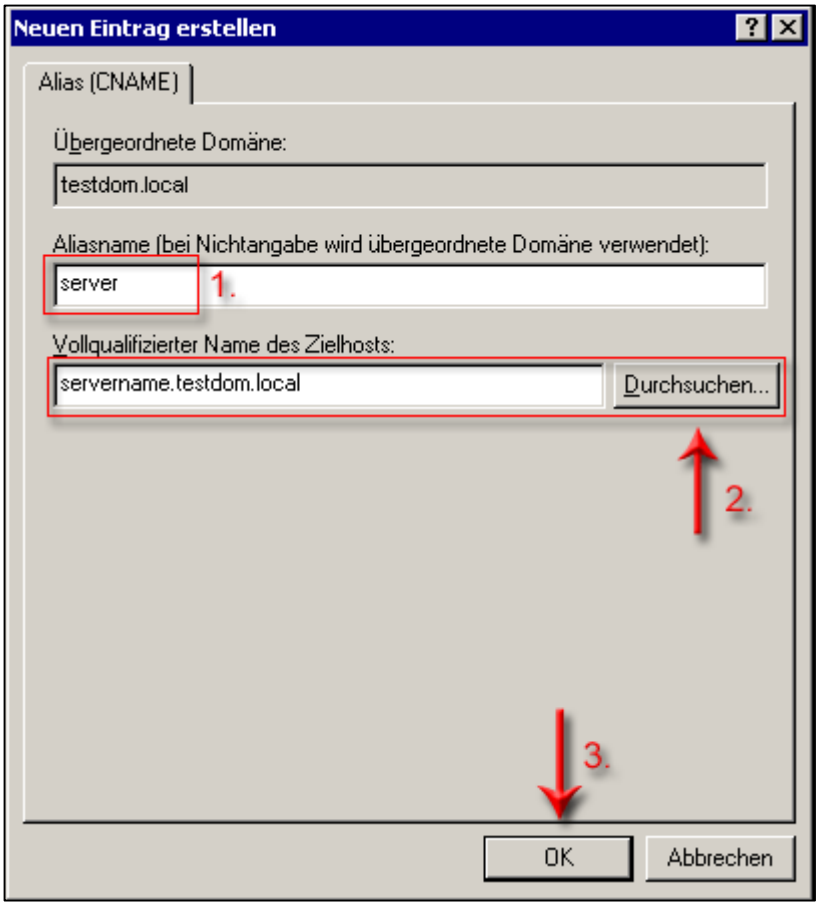
Zum Schluss wird der Assistent mit `Fertig stellen` beendet. Die Konfiguration der *Reverse-Lookupzone* wird mit der Auswahl `Neue Zone...` aus dem Kontextmenü der `Reverse-Lookupzone` eingeleitet. Das erste Fenster des Assistenten wird wie gewohnt mit `Weiter` bestätigt. Alle weiteren Fenster sind identisch mit dem obigen Verfahren. Nur bei der `Netzwerkkennung` sollte die Netzklasse angegeben werden (z.B. 192.168.120.).



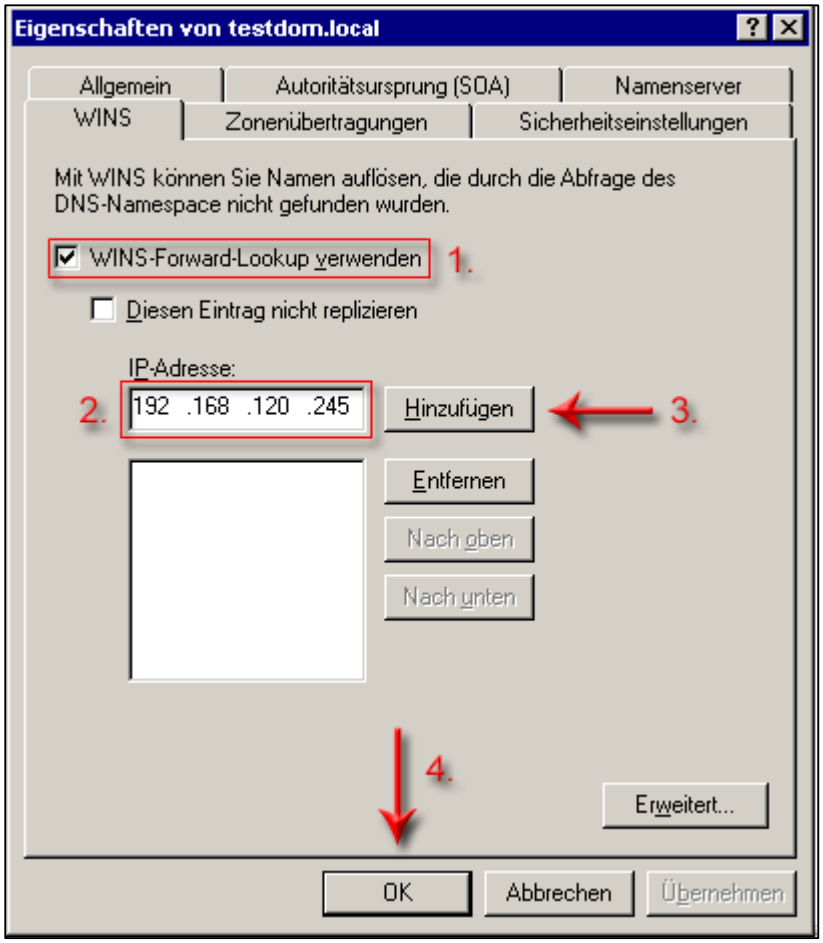
Mit einem Rechtsklick auf die soeben erstellte `Forward-Lookupzone` sowie `Reverse-Lookupzone` können neue *Hosts*, *Aliase*, *Mail-Exchanger*, *Domänen*, *Delegierungen* und noch viel mehr in die

Liste eingepflegt werden. Ich werde hier nur auf die *Hosts* und *Aliase* eingehen, weil sonst dieser Artikel das Installationshandbuch sprengen würde. Am besten werden *Hosts* in der *Fordward-Lookupzone* erstellt, da dort die Möglichkeit besteht die Informationen an die *Reverse-Lookupzone* gleich weiterzugeben. Dies geschieht durch die Option `Verknüpften PTR-Eintrag erstellen`. Ein *Alias* wird immer erst nach einem *Host* angelegt, da dieser wahrscheinlich auf den angelegten *Host* verweisen soll. Wahlweise kann der *Host* über die `Durchsuchen` Option aufgesucht werden, oder er wird komplett manuell eingetragen.



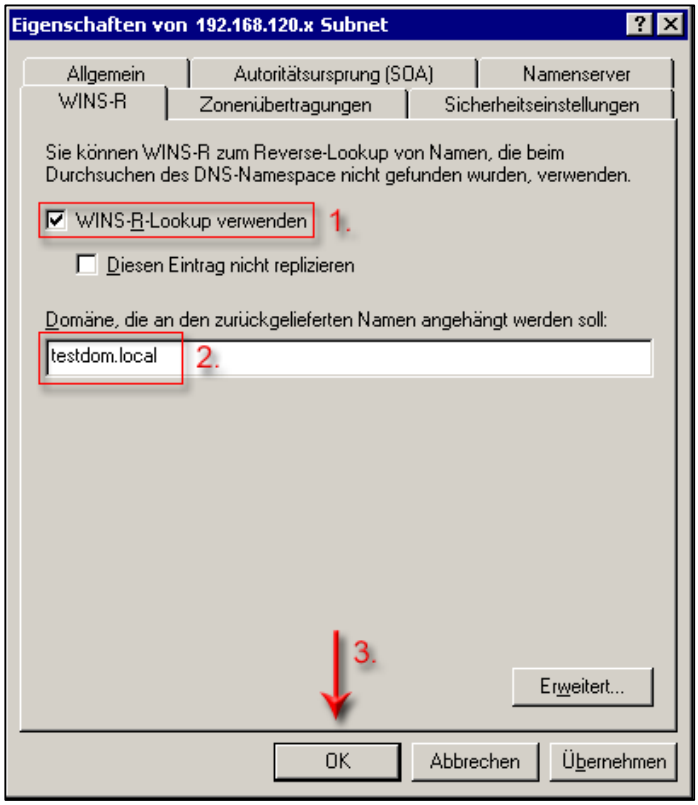


Wichtig ist der Verweis auf einen vorhanden *WINS-Server* im lokalen Netzwerk, wenn einer installiert wurde. Die Konfiguration muss bei beiden eingerichteten Lookupzonen geschehen. Bei der *Fordward-Lookupzone* wird die Baumstruktur erweitert und ein Rechtsklick auf `testdom.local` ausgeführt. Im erscheinenden Kontextmenü wählt man die `Eigenschaften` aus und wechselt auf die Reiterkarte `WINS`.

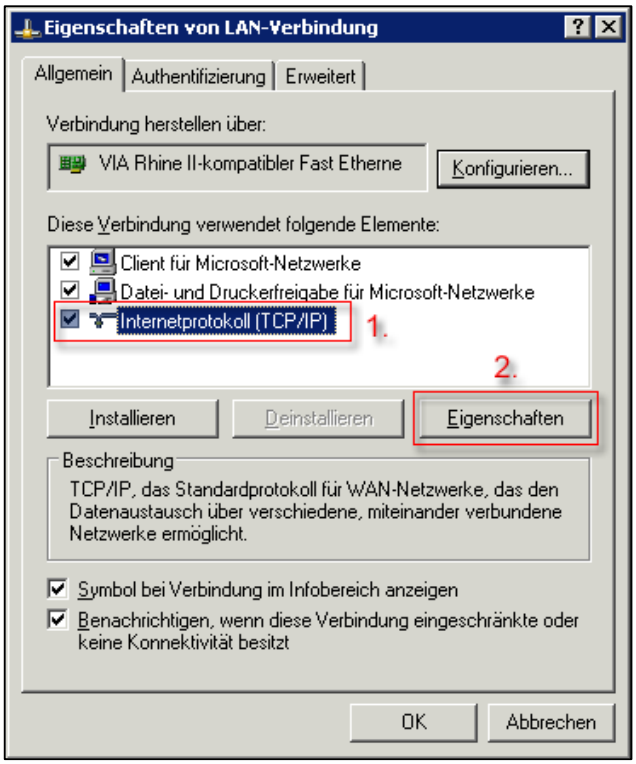


Dort wird der Hacken bei `WINS-Forward-Lookup` verwenden gesetzt. Dadurch wird die Möglichkeit eine `IP-Adresse` des WINS-Servers einzugeben frei geschaltet. Hier sollte die `IP-Adresse` mit dem `Hinzufügen` Knopf übernommen werden. Natürlich ist es auch hier möglich mehrere *WINS-Server* anzugeben.

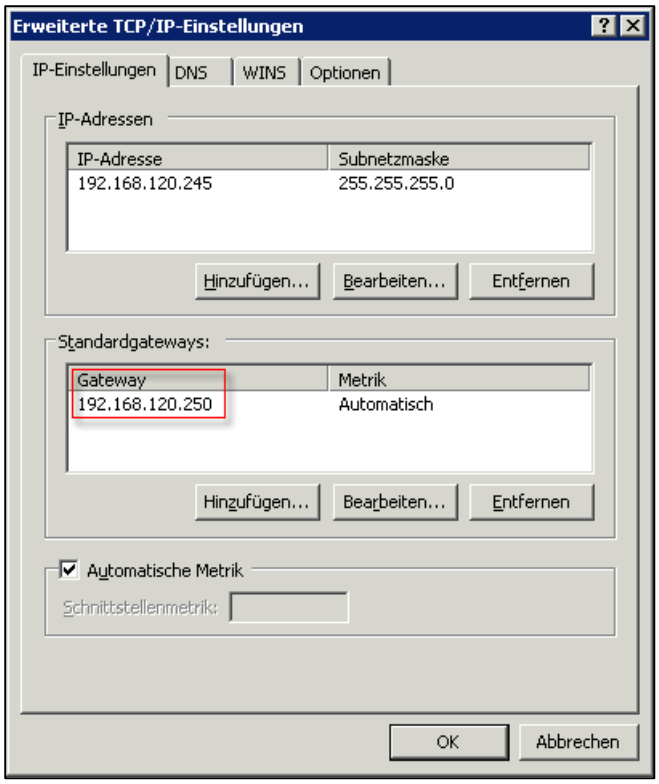
Das gleiche wird bei der *Reverse-Lookupzone* konfiguriert. Unter der Reiterkarte `WINS-R` sollte der Hacken bei `WINS-R-Lookup verwenden` gesetzt werden. Danach wird die Option eine Domäne anzugeben frei geschaltet. Die Domäne `testdom.local` wird eingetragen. Abschließend wird noch mit `OK` bestätigt.



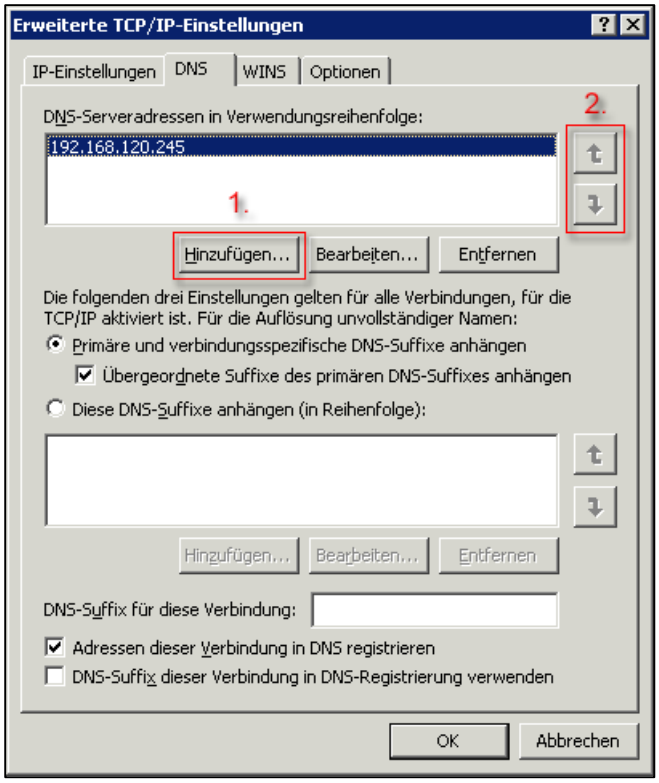
Der *DNS-Server* ist soweit in Bezug auf die Zonen richtig konfiguriert, aber die Eigenschaften des Servers selber fehlen und eine Anfrage wird derzeit noch nicht richtig aufgelöst, da die Eigenschaften der Netzwerkverbindungen noch auf die *Loopback* Adresse des Servers verweisen. Erster Schritt ist die `Eigenschaften` der `Netzwerkumgebung` aufzurufen und dann die `Eigenschaften` der `LAN-Verbindung`. In diesem Fenster sollte unter der Reiterkarte `Allgemein` das Protokoll *TCP/IP* ausgewählt werden. Per Mausklick auf `Eigenschaften` können detaillierte Informationen geändert.



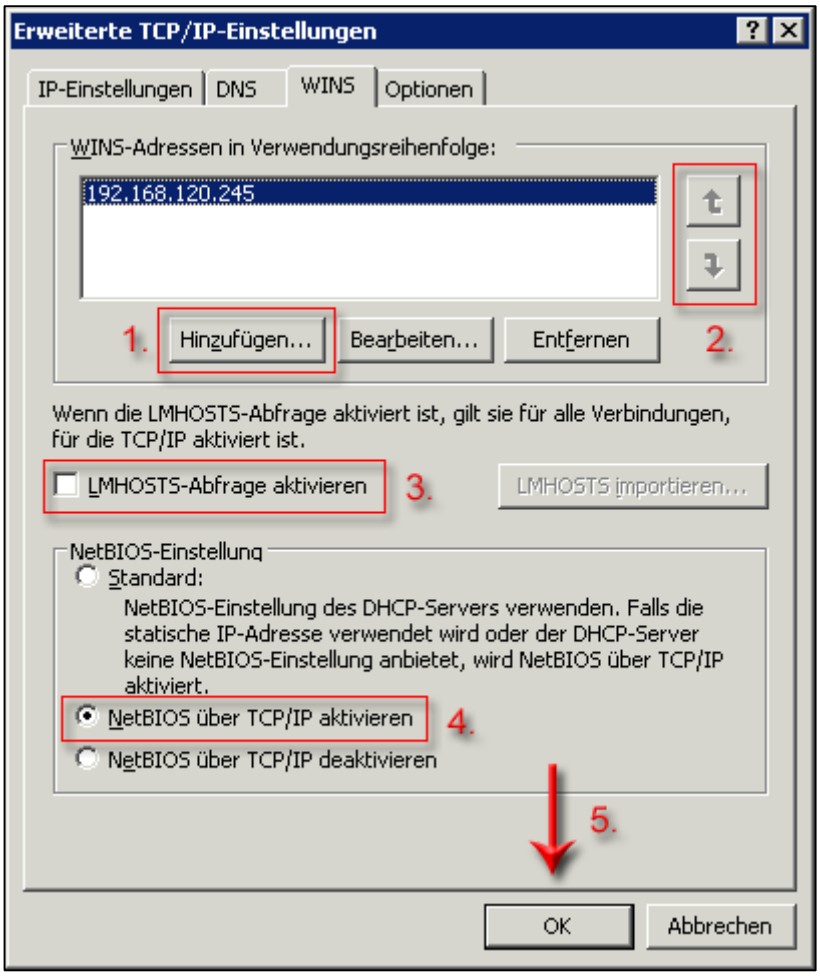
Unter den `Eigenschaften` wählt man `Erweitert` aus. Die Reiterkarte `IP-Einstellungen` sollte noch ergänzt werden. Wenn sich ein *Router* im Netzwerk befindet, sollte dieser unter `Gateway` hinzugefügt werden.



Unter der Reiterkarte `DNS` sollte unbedingt der *DNS-Server* eingepflegt werden. Dies kann per Mausklick auf `Hinzufügen` geschehen. Des Weiteren ist es nötig, dass der interne *DNS-Server* (lokaler Server) an erster stelle steht, damit alle Anfragen vom Server als erstes an den lokalen *DNS* gesendet werden. Darüber hinaus ist es Sinnvoll auch noch die *DNS-Server* seines *ISP's* anzugeben. Diese sollten aber erst am Ende der Liste auftauchen.

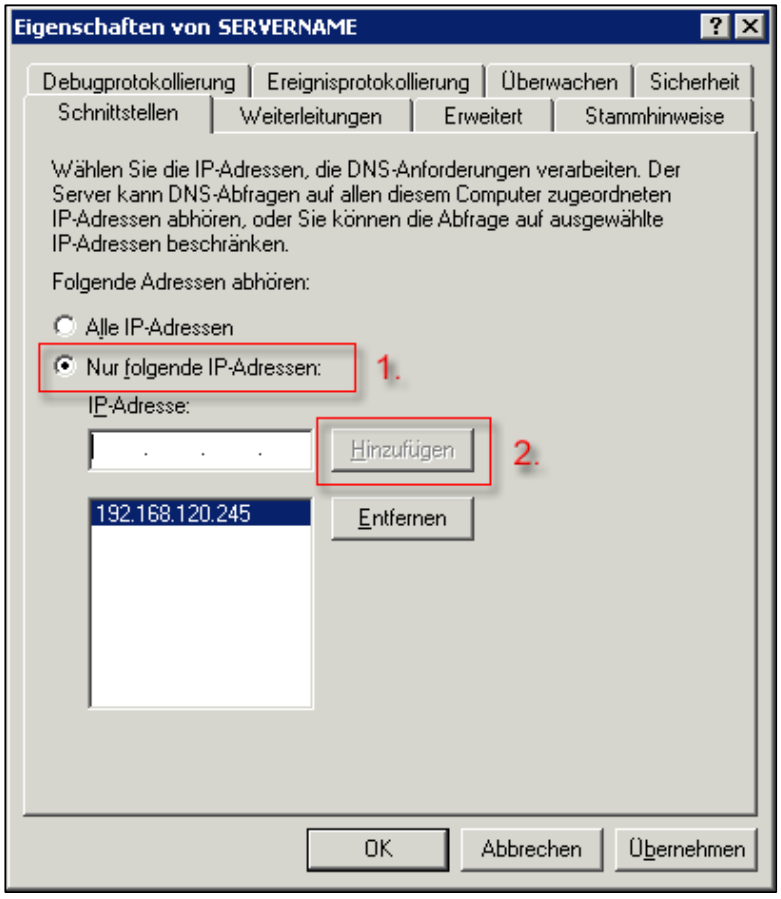


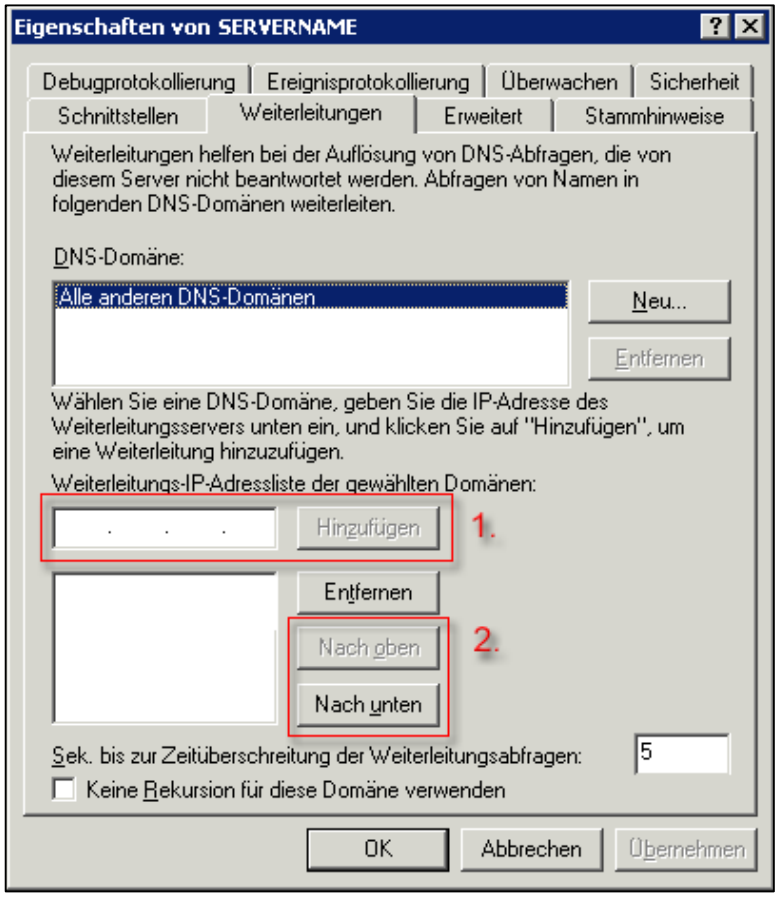
Bei einem *WINS-Server* im lokalen Netzwerk empfiehlt es sich diesen unter der Reiterkarte `WINS` einzupflegen. Sollte es weitere *WINS-Server* geben, können diese wahlweise ebenfalls angegeben werden. Wichtig ist nur, dass der Server eine statische IP-Adresse hat und somit keine Informationen von einem *DHCP-Server* empfangen sollte. Aus diesem Grund setzt man die *NetBIOS* Einstellungen auf aktiviert (`NetBIOS über TCP/IP aktivieren`).

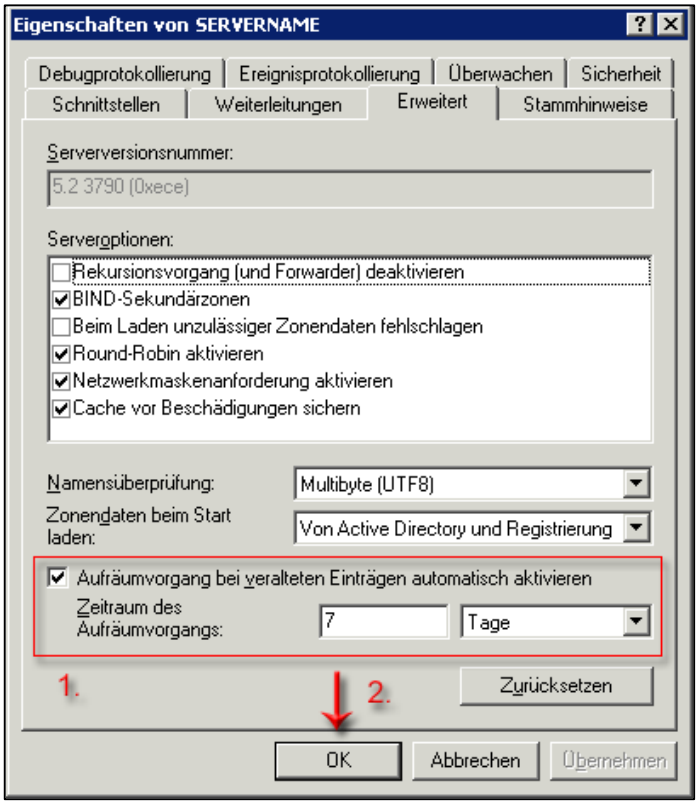


Die `LMHOSTS-Abfrage` sollte deaktiviert werden, da der Server nicht in seiner lokalen `lmhosts.sam` wegen der Namensauflösung nachforschen soll.

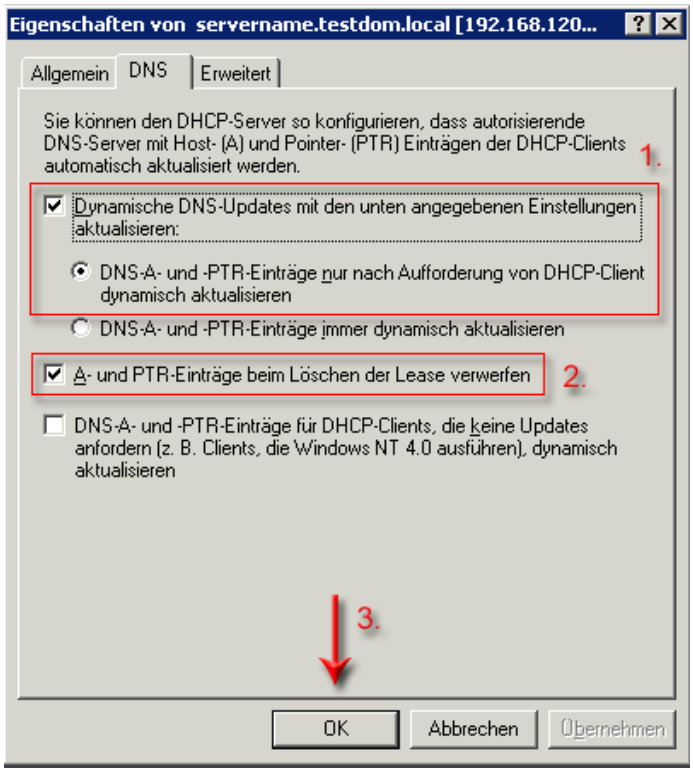
Diese Einstellungen müssen an jedem Client im Netzwerk getätigt werden, wenn kein *DHCP* eingesetzt wird, sprich statische IP-Adressen verwendet werden. Jedoch sind das noch nicht alle Schritte die zum erfolgreichen Einsatz des *DNS-Servers* benötigt werden. Zum Schluss ist noch die Eigenschaft des Servers selber zu konfigurieren. Dies geschieht durch den Aufruf des `DNS` Snap-In's und einen Rechtsklick auf den Server selber um im Kontextmenü die `Eigenschaften` anzuwählen. Auf der Reiterkarte `Schnittstellen` sollte von der Option `Alle IP-Adressen` auf `Nur folgende IP-Adressen:` verwenden gewechselt werden, damit der lokale *DNS-Server* alleinig angefragt werden kann. Wahlweise können natürlich mehrer *DNS-Server* im lokalen Netzwerk angeben werden. Im zweiten Schritt wird auf die Reiterkarte `Weiterleitungen` geklickt. Hier sollten die *DNS-Server* des *ISP's* angegeben werden. Damit ist klar und deutlich gesagt, dass alle gestellten Anfragen erstmal an den lokalen *DNS-Server* gehen, kann dieser die Anfrage nicht auflösen, wird sie an die *DNS-Server* des Providers weitergeleitet. Bei Bedarf könnte jetzt noch unter `Erweitert` die `Aufräumvorgänge` aktiviert werden. Der Zeitraum ist an die Bedürfnisse anzupassen.


Eigenschaften von SERVERNAME
Debugprotokollierung
Ereignisprotokollierung
Überwachen
Sicherheit
Schnittstellen
Weiterleitungen
Erweitert
Stammhinweise
Wählen Sie die IP-Adressen, die DNS-Anforderungen verarbeiten. Der Server kann DNS-Abfragen auf allen diesem Computer zugeordneten IP-Adressen abhören, oder Sie können die Abfrage auf ausgewählte IP-Adressen beschränken.
Folgende Adressen abhören:
Alle IP-Adressen
Nur folgende IP-Adressen:
1.
IP-Adresse:
Hinzufügen
2.
192.168.120.245
Entfernen
OK
Abbrechen
Übernehmen


Eigenschaften von SERVERNAME
Debugprotokollierung
Ereignisprotokollierung
Überwachen
Sicherheit
Schnittstellen
Weiterleitungen
Erweitert
Stammhinweise
Weiterleitungen helfen bei der Auflösung von DNS-Abfragen, die von diesem Server nicht beantwortet werden. Abfragen von Namen in folgenden DNS-Domänen weiterleiten.
DNS-Domäne:
Alle anderen DNS-Domänen
Neu...
Entfernen
Wählen Sie eine DNS-Domäne, geben Sie die IP-Adresse des Weiterleitungsservers unten ein, und klicken Sie auf "Hinzufügen", um eine Weiterleitung hinzuzufügen.
Weiterleitungs-IP-Adressliste der gewählten Domänen:
Hinzufügen
1.
Entfernen
Nach oben
Nach unten
2.
Sek. bis zur Zeitüberschreitung der Weiterleitungsabfragen:
5
Keine Rekursion für diese Domäne verwenden
OK
Abbrechen
Übernehmen



Natürlich ist die manuelle Pflege eines *DNS-Servers* in Kombination mit einem *DHCP-Server* nicht nötig. Bei dieser Konfiguration sollte die Reiterkarte `DNS` unter den `Eigenschaften` des `DHCP` Snap-In's aufgerufen werden. Dort sollte `Dynamische DNS-Updates mit den unten angegebenen Einstellungen aktualisieren` sowie der Unterpunkt `DNS-A- und PTR-Einträge nur nach Aufforderung von DHCP-Client dynamisch aktualisieren` aktiviert werden. Des Weiteren ist es von nutzen den Hacken bei `A- und PTR-Einträge beim Löschen des Lease verwerfen` zu setzen. Abschließend kann die Konfiguration mit `OK` beendet werden.



Microsoft Exchange Server

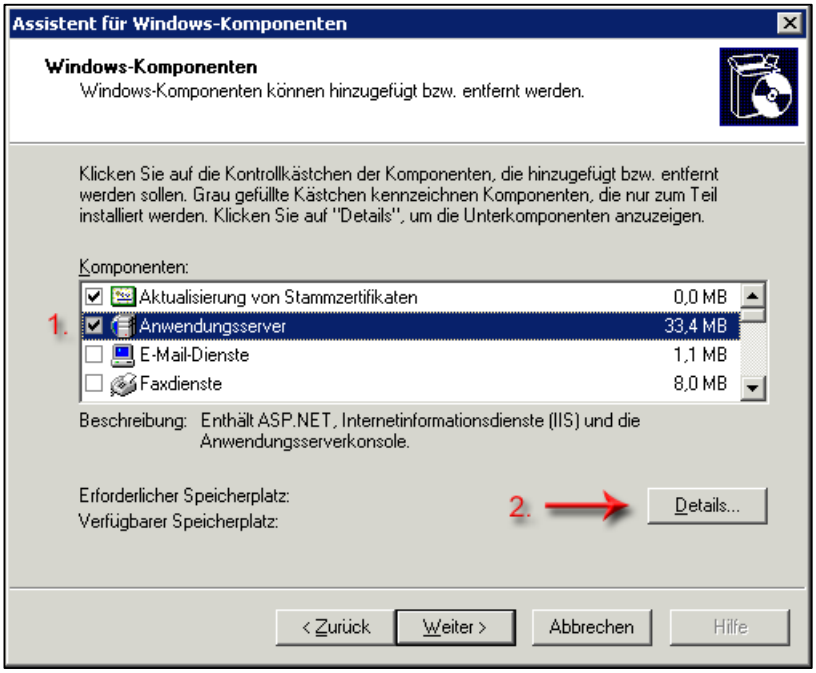
Der *Microsoft Exchange Server* ist ein komplexer Mailserver und die Konfiguration ist ein wenig zeitaufwändig. Sicherlich ist es möglich viele wichtige Installations- und Konfigurationsschritte zu erklären, jedoch gibt es schon eine ausgezeichnete Website über die Installation, Konfiguration, Probleme und Notfallsituationen des Exchange Servers von Exchange 5.x bis zum Exchange 2003. Deshalb habe ich mich entschieden hier noch mal den Link auf die Website von Herrn Frank Carius zu veröffentlichen (http://www.msxfaq.de). Unter meiner Linkrubrik ist der Link ebenfalls zu finden. Für mich ist es blödsinnig in ähnlicher Form die Informationen nochmals zu veröffentlichen, jedoch soll dies nur ein Installationsleitfaden sein und keine detaillierte Auskunft über den *Microsoft Exchange Server*.

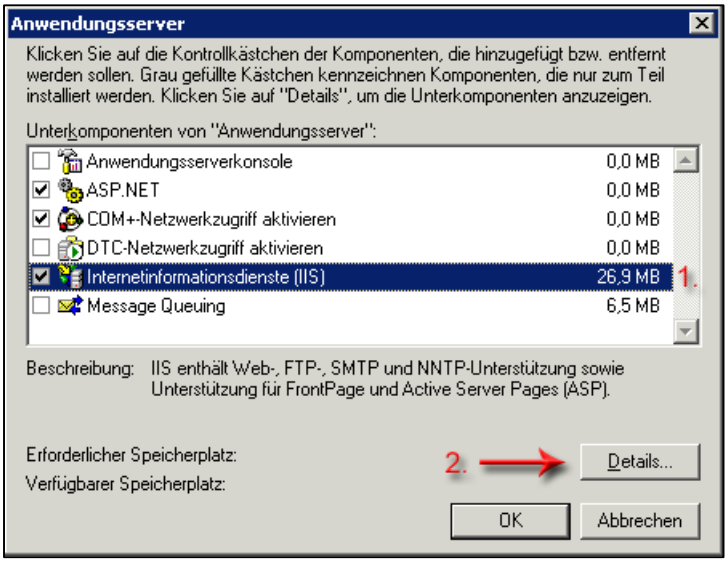
Einrichtung und Verwendung des Software Update Services

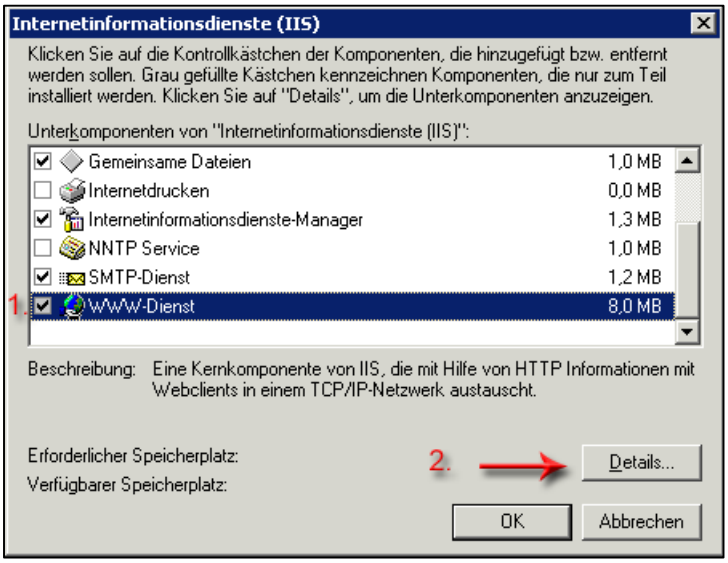
Download und Installation des SUS-Servers

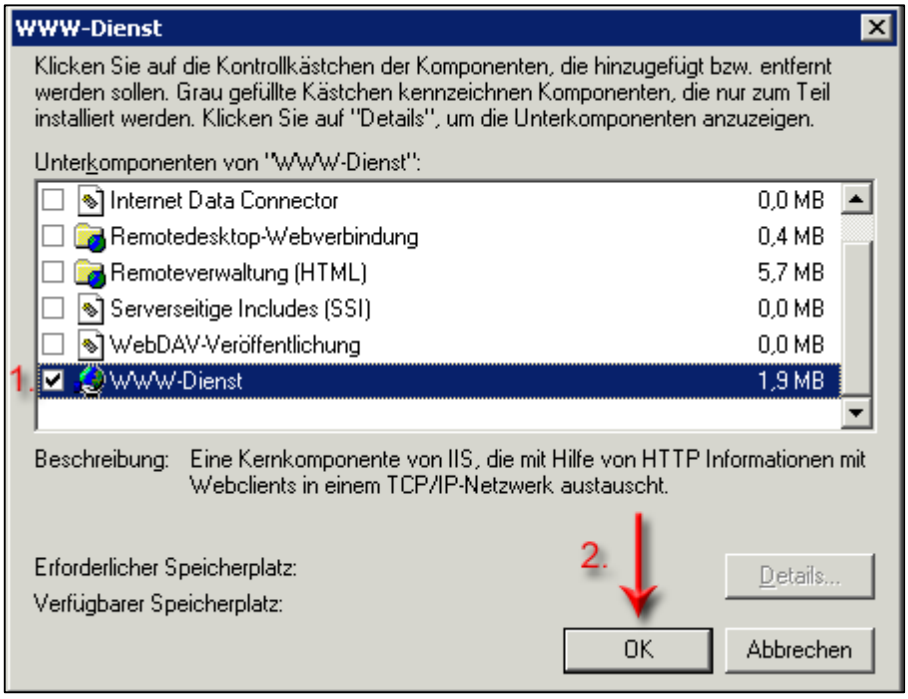
Der *Software Update Services* dient dazu, die Arbeitsstationen in einem lokalen Netzwerk mit kritischen Updates von *Microsoft* zu bestücken. Keine Firma möchte gerne die Schnittstelle zum öffentlichen Netz so sehr auslasten, dass kaum eine Website mehr angezeigt werden kann (Bandbreite ausgelastet). Dies geschieht durch Arbeitsstationen, die gerade versuchen kritische Updates vom *Microsoft* Update Server herunter zu laden. Aus diesem Grund empfiehlt es sich im lokalen Netzwerk einen SUS-Server einzusetzen, da dieser gezielt zu Ruhezeiten, Updates herunterladen kann. Sicherlich es ebenso möglich für Arbeitsstationen Updates in Ruhezeiten einzuspielen, jedoch müsste dazu jeder Rechner in der Nacht laufen, weil Ruhezeiten einer Firma nachts ist. Das hätte zur Folge, dass der Stromverbrauch fürs Unternehmen immens steigen würde. Der SUS-Server wird kostenlos von *Microsoft* angeboten, leider aber nur in den Sprachen Englisch und Japanisch. Für den europäischen Markt empfiehlt sich die englischsprachige Version. Unter der Rubrik Downloads auf meiner Website biete ich den kostenlosen SUS-Server in der Version 1.0 an.

Die Voraussetzungen für eine erfolgreiche Installation sind denkbar einfach. Lediglich der *Internet Information Services* muss installiert werden. Per Mausklick auf `Software` unter `Start` **à** `Einstellungen` **à** `Systemsteuerung` **à** `Software` ist es möglich in der Rubrik `Windows Komponenten hinzufügen/entfernen` den *IIS* Dienst zu installieren bei einem *Microsoft Windows 2003 Server*. Im ersten Fenster markiert man den `Anwendungsserver` und wechselt per Mausklick auf `Details…` in das sich öffnende Fenster `Anwendungsserver`. Die Markierung der `Internetinformationsdienste (IIS)` und einem erneuten Mausklick auf `Details…` leitet einen zum neuen Fenster `Internetinformationsdienste (IIS)`. Durch die Markierung der `WWW-Dienste` und dessen Bestätigung durch `Details…` öffnet das Fenster `WWW-Dienste`. Abschließend wählt man den `WWW-Dienst` aus (Häkchen setzen) und beendet mit einem Mausklick auf `OK`. Nach dem Kopiervorgang muss der Server neu gestartet werden.

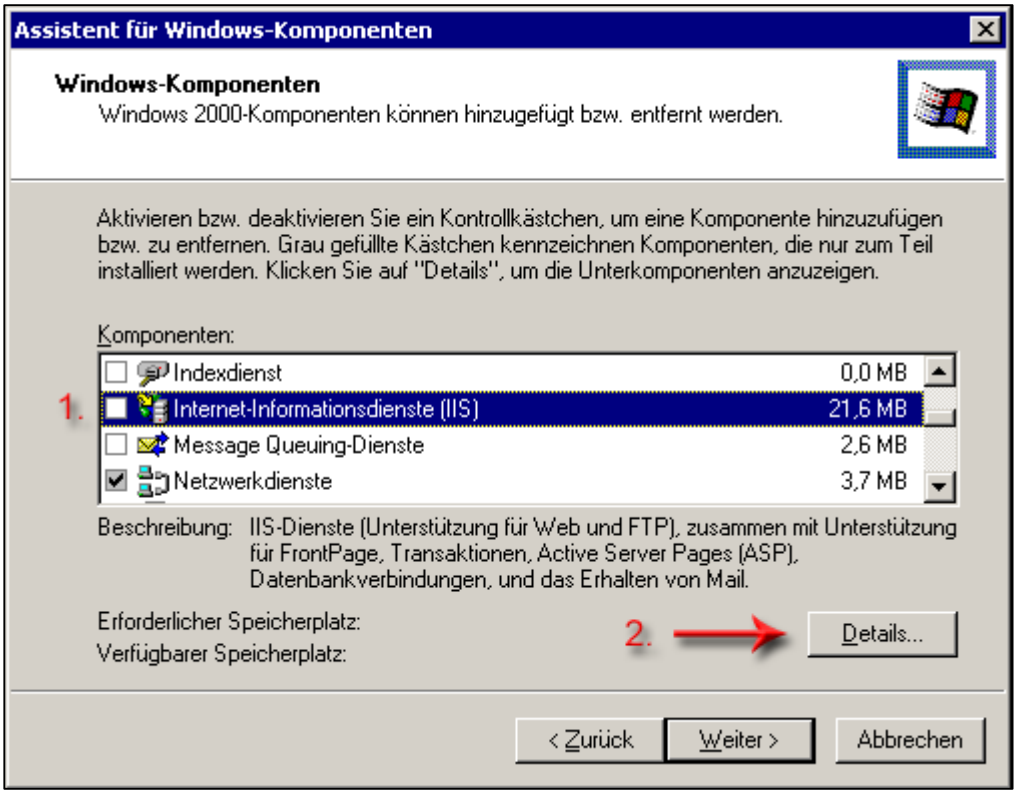

Assistent für Windows-Komponenten
Windows-Komponenten
Windows-Komponenten können hinzugefügt bzw. entfernt werden.
Klicken Sie auf die Kontrollkästchen der Komponenten, die hinzugefügt bzw. entfernt werden sollen. Grau gefüllte Kästchen kennzeichnen Komponenten, die nur zum Teil installiert werden. Klicken Sie auf "Details", um die Unterkomponenten anzuzeigen.
Komponenten:
Aktualisierung von Stammzertifikaten 0,0 MB
1.
Anwendungsserver 33,4 MB
E-Mail-Dienste 1,1 MB
Faxdienste 8,0 MB
Beschreibung: Enthält ASP.NET, Internetinformationsdienste (IIS) und die Anwendungsserverkonsole.
Erforderlicher Speicherplatz:
Verfügbarer Speicherplatz:
2.
Details...
< Zurück
Weiter >
Abbrechen
Hilfe

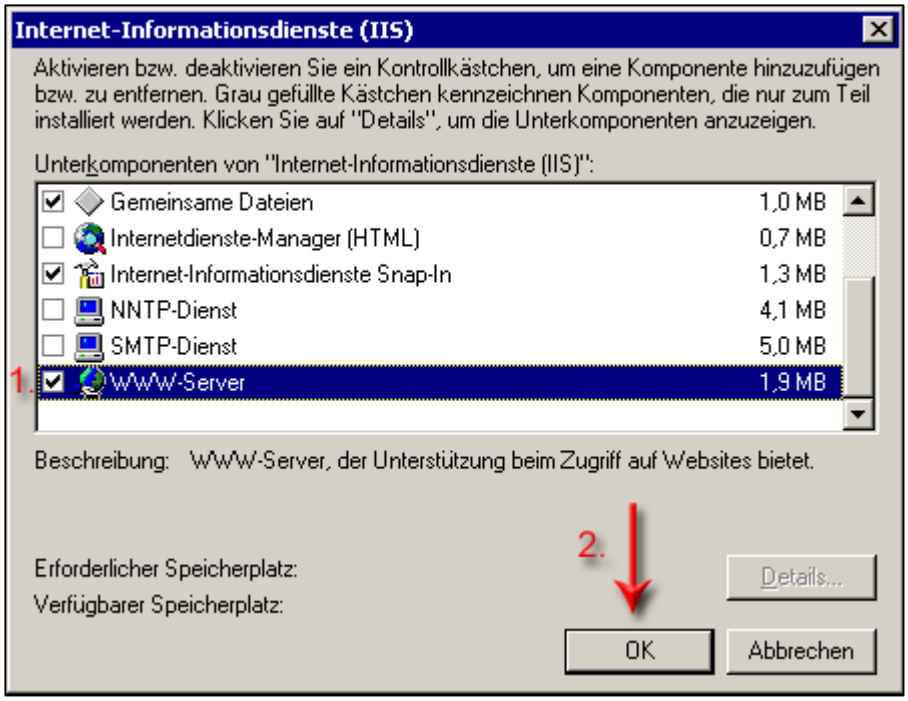

Anwendungsserver
Klicken Sie auf die Kontrollkästchen der Komponenten, die hinzugefügt bzw. entfernt werden sollen. Grau gefüllte Kästchen kennzeichnen Komponenten, die nur zum Teil installiert werden. Klicken Sie auf "Details", um die Unterkomponenten anzuzeigen.
Unterkomponenten von "Anwendungsserver":
Anwendungsserverkonsole 0,0 MB
ASP.NET 0,0 MB
COM+-Netzwerkzugriff aktivieren 0,0 MB
DTC-Netzwerkzugriff aktivieren 0,0 MB
Internetinformationsdienste (IIS) 26,9 MB
1.
Message Queuing 6,5 MB
Beschreibung: IIS enthält Web-, FTP-, SMTP und NNTP-Unterstützung sowie Unterstützung für FrontPage und Active Server Pages (ASP).
Erforderlicher Speicherplatz:
Verfügbarer Speicherplatz:
2.
Details...
OK
Abbrechen


Internetinformationsdienste (IIS)
Klicken Sie auf die Kontrollkästchen der Komponenten, die hinzugefügt bzw. entfernt werden sollen. Grau gefüllte Kästchen kennzeichnen Komponenten, die nur zum Teil installiert werden. Klicken Sie auf "Details", um die Unterkomponenten anzuzeigen.
Unterkomponenten von "Internetinformationsdienste (IIS)":
Gemeinsame Dateien 1,0 MB
Internetdrucken 0,0 MB
Internetinformationsdienste-Manager 1,3 MB
NNTP Service 1,0 MB
SMTP-Dienst 1,2 MB
1.
WWW-Dienst 8,0 MB
Beschreibung: Eine Kernkomponente von IIS, die mit Hilfe von HTTP Informationen mit Webclients in einem TCP/IP-Netzwerk austauscht.
Erforderlicher Speicherplatz:
Verfügbarer Speicherplatz:
2.
Details...
OK
Abbrechen



Bei einem *Microsoft Windows 2000 Server* sieht das ganze etwas anders aus. Die Schritte bis zum Fenster `Assistent für Windows-komponenten` bleibt wie oben beschrieben gleich. Der `Internet-Informationsdienst (IIS)` wird markiert und per Mausklick auf `Details`… gelangt man in das Fenster `Internet-Informationsdienste (IIS)`. Abschließend wird hier der `WWW-Server` mit einem Häkchen versehen und per Mausklick auf `OK` bestätigt. Nach dem Kopiervorgang muss der Server neu gestartet werden.

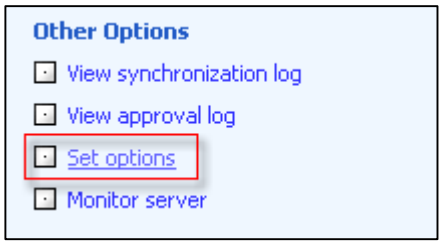




Nach erfolgter Einrichtung der Webserverdienste auf dem Server kann nun der SUS-Sever installiert werden. Per Doppelklick auf die Setupdatei wird die Installationsroutine gestartet. Das erste Fenster kann wie immer mit `Next` bestätigt werden. Des Weiteren müssen die Lizenzbedienungen (*EULA*) anerkannt werden, weil sonst die Installation abgebrochen wird. Dies geschieht durch das setzen des Hackens bei `I accept…` und einem Mausklick auf `Next`. Der Installationstyp sollte mit `Typical` bestätigt werden. Zum Schluss muss nur noch mit `Install` der Installationsvorgang eingeleitet und nach Abschluss mit `Finish` beendet werden.

Konfiguration des SUS-Servers

Die Konfiguration des SUS-Servers ist denkbar einfach. Man öffnet den *Internet Explorer* und gibt in der Adresszeile `http://localhost/susadmin` ein. Es erscheint die Konfigurationsseite des SUS-Servers, die an die Version 4 der Windows Update Seite erinnert. Unter `Other Options` wird `Set Options` ausgewählt.



Damit der SUS-Server seine Updates von der *Microsoft* Website beziehen kann, ist es nötig die Verbindung zum öffentlichen Netz anzugeben. Hier ist es möglich einen *Proxy-Server*, oder einfach `Do not use a proxy server to access to the Internet` bei einem *Router* anzugeben. Als nächstes muss ein Name für den lokalen SUS-Server angegeben werden. Hier kann man entscheiden, ob man nur den *NetBIOS*-, oder den kompletten *DNS*-Namen verwendet. Falls sich Arbeitsstationen im Netzwerk befindet, die keine *NetBIOS* Unterstützung haben, sollte man auf jeden Fall den *Full Qualified Domain Name* (*FQDN*) angeben. Im nächsten Schritt wird gefragt, ob man direkt die Updates vom *Microsoft* Update Server, oder einem anderen Update Server beziehen möchte. Hier ist es auf jeden Fall sinnvoll den *Microsoft* Server zu verwendet. Abschließend ist es möglich die Updates direkt vom *Microsoft* Update Server weiterzuleiten (keine lokale Speicherung), oder diese auf die lokale Festplatte zu speichern. Falls nicht genügend Festplattenspeicher zur Verfügung steht, ist es ratsam direkt von der *Microsoft* Website die Updates zu beziehen. Wenn man aber genügend Speicherkapazität hat, ist es empfehlenswert die Updates lokal zu speichern. Schließlich ist die Übertragungs-

geschwindigkeit um einiges höher im lokalen Netzwerk. Zusätzlich kann man den Festplattenspeicherbedarf drosseln, indem man einfach nur die benötigte Sprache auswählt. Mit `Apply` werden die Änderungen gespeichert.

Vorletzter serverseitiger Arbeitsschnitt ist, die Synchronisation zu planen. Per Auswahl von `Synchronize server` ist es möglich den Synchronisationszeitraum zu planen. Der Zeitraum sollte in die Ruhezeiten, sprich nachts, fallen da dort mit Sicherheit keiner die Bandbreite der öffentlichen Schnittstelle ausnutzen möchte. Nach einer erfolgreichen Synchronisation müssen die entsprechenden Updates freigegeben werden. Dies ist eine ziemlich nervenaufreibende Tätigkeit, da jedes Update einzeln freigegeben werden muss und es mehrere hundert Updates sind. Dieser Schritt geschieht unter `Approve Updates`.

Automatisierung der Freigabe von Updates

Um sich diese Arbeit zu erleichtern gibt es ein schönes *Visual Basic Script* das nur noch richtig konfiguriert werden muss. Um das Script zu erstellen legt man eine neue `Textdatei` mit beliebigem Namen an und ändert die Dateierweiterung von `.txt` in `.vbs` um. Um den Inhalt bearbeitet zu können wird per rechten Mausklick auf die Datei das Kontextmenü geöffnet und dort `Bearbeiten` angewählt. Jetzt kopiert man den folgenden Inhalt hinein, oder ladet die Visual Basic Script Datei unter meiner Downloadrubrik herunter.

```
''Auto Approve SUS Patches
''
''To use:
''Set up a scheduled job to run "cscript c:\autoapproveupdates.vbs" and
run with admin rights
''

'' SETUP Section
'' ------------------------
''Use default C on local machine

strSUSpath = "C:"

''Uncomment the line below and insert your server name if you want to run
remotely
''strSUSpath = "\\servername\c$"

'' If true will not approve XP SP2

nosp2 = false

'' If true will not approve Windows 2003 server SP1

noWS03SP1 = true

'' If true will not approve Windows 2000 service pack 4

noW2KSP4 = false

'' If false will use CDonts

usejmail = false
```

```
'' Email configuration

EmailDstName = "name@sld.tld"
EmailReplyToName = "name@sld.tld"

'' Can use localhost if you have an smtp server locally

EmailSrvName = "smtpserver.sld.tld"


'' Start of Code
'' -------------------------

debugflag = true
NumNewPatches = 0

Const ForReading = 1       ''Open a file for reading only. You can't wri-
te to this file.
Const ForWriting = 2       ''Open a file for writing. If a file with the
same name exists, its previous contents is overwritten.
Const ForAppending = 8     ''Open a file and write to the end of the fi-
le.
Const TristateUseDefault = -2    ''Opens the file using the system de-
fault.
Const TristateTrue = -1    ''Opens the file as Unicode.
Const TristateFalse = 0    ''Opens the file as ASCII.

Set objFileSystem = CreateObject("Scripting.FileSystemObject")

strPath = strSUSpath &
"\Inetpub\wwwroot\autoupdate\dictionaries\ApprovedItems.txt"

strappenddate = year(now()) & month(now()) & day(now()) & hour(Now()) &
minute(now())
strPathBackup = strSUSpath &
"\Inetpub\wwwroot\autoupdate\dictionaries\ApprovedItems_"& strappenddate
&".txt"

''Opens the input file
set objApprovalfile = objFileSystem.GetFile(strPath)
set txtStream = objApprovalfile.OpenAsTextStream(ForReading, TristateUse-
Default)
strApprovalContents = txtStream.ReadAll
txtStream.close()
Set objApprovalfile = Nothing

LogIt("Opened Approvedupdates.txt")

''Backup the previous ApprovedItems.txt file
set CopyobjApprovalfile = objFileSystem.GetFile(strPath)
CopyobjApprovalfile.Copy(strPathBackup)
Set CopyobjApprovalfile = Nothing

'Parse the new patches description
set objRegEx = New RegExp
set ParsePatchName = New RegExp
objRegEx.Global = True
ParsePatchName.Global = False
```

```
objRegEx.Pattern = "com_microsoft.[A-z0-9_\.\s]*,[0 2]@\|"
ParsePatchName.Pattern = "\.[A-z0-9_\s]*,"
Set Matches = objRegEx.Execute(strApprovalContents)
For Each Match in Matches
    Set ParseMatch = ParsePatchName.Execute(Match.Value)
    StrTemp = ParseMatch.Item(0).Value

    StrTemp = Replace(StrTemp, ".", "")
    StrTemp = Replace(StrTemp, ",", "")

    Select case StrTemp
          case "xp_sp_2"
                If not nosp2 then strReturnStr = increment_NumNewPatches
()
          case "ws03_sp1_sus"
                If not noWS03SP1 then strReturnStr = incre-
ment_NumNewPatches ()
          case "windows 2000 service pack 4 network install for it pro-
fessionals"
                If not noW2KSP4   then strReturnStr = incre-
ment_NumNewPatches ()
          case else
                strReturnStr = increment_NumNewPatches ()
    End select
Next

Function increment_NumNewPatches ()
    StrArray = Split(StrTemp, "_")
    NumKeywords = Ubound(StrArray)

    If NumKeywords  >= 0 Then
          increment_NumNewPatches = increment_NumNewPatches & VBNewLine &
"" & StrArray(0) & " : "
          If NumKeywords >= 2 Then
                For k = 1 To NumKeywords
                      increment_NumNewPatches = increment_NumNewPatches &
StrArray(k) & " "
                Next
          End If
    Else
          increment_NumNewPatches = increment_NumNewPatches & VBNewLine &
"Unknown patch name format :" & StrTemp
    End If
    NumNewPatches = NumNewPatches+1
End function

Set objRegEx = Nothing
Set ParsePatchName= Nothing


'' Approve patches

strApprovalContents = Replace(strApprovalContents, ",0@|", ",1@|")
strApprovalContents = Replace(strApprovalContents, ",2@|", ",1@|")

If (nosp2) Then
    strApprovalContents = Replace(strApprovalContents, "xp_sp_2,1@|",
"xp_sp_2,0@|")
End If
```

```
if (noWS03SP1) Then
    strApprovalContents = Replace(strApprovalContents,
"ws03_sp1_sus,1@|", "ws03_sp1_sus,0@|")
End If

if (noW2KSP4) Then
    strApprovalContents = Replace(strApprovalContents, "windows 2000 ser-
vice pack 4 network install for it professionals,1@|", "windows 2000 ser-
vice pack 4 network install for it professionals,0@|")
End If

LogIt(strApprovalContents)

'' Write file back to disk
set objApprovalfile = objFileSystem.GetFile(strPath)
set txtStream = objApprovalfile.OpenAsTextStream(ForWriting, TristateUse-
Default)
txtStream.Write(strApprovalContents)
txtStream.Close()

Set objApprovalfile = Nothing

LogIt("File updated")

'' Send email if new patches are approved
if (NumNewPatches > 0) then

    if (NumNewPatches > 1) then strPlural="es"

    strMsgBody =      "New patch" & strPlural & " approved: (" & Now &
")" & VBNewLine & _
          "-------------------------------------------" & VBCR & _
          strReturnStr


    if (usejmail) then

          ' Create the JMail message Object
          set msg = CreateOBject("JMail.Message")

          msg.Logging = true
          msg.silent = true
          msg.From = EmailReplyToName
          msg.AddRecipient(EmailDstName)
          msg.Subject = "[SUS Server] " & NumNewPatches & " New Patch" &
strPlural & " Approved"
          msg.Body = strMsgBody

          if (msg.Send(EmailSrvName)) then
                LogIt("Success sent:" & vbNewline & strMsgBody)
                Logit(msg.log)
          else
                LogIt("Error sending email!")
                Logit(msg.log)
          end if

    else

          Set iMsg = CreateObject("CDO.Message")
```

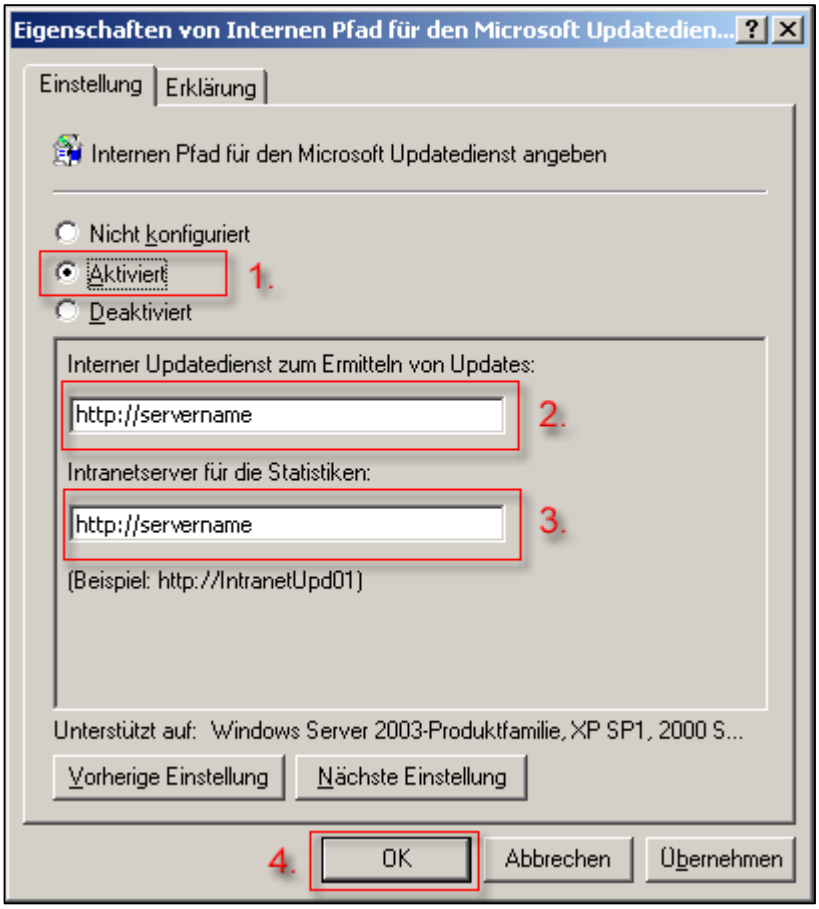
```
    iMsg.Configuration.Fields.Item("http://schemas.microsoft.com/cdo/conf
iguration/smtpserver") = EmailSrvName

    iMsg.Configuration.Fields.Item("http://schemas.microsoft.com/cdo/conf
iguration/smtpserverport") = 25

    iMsg.Configuration.Fields.Item("http://schemas.microsoft.com/cdo/conf
iguration/sendusing") = 2

          '' Uncomment these and set for authenticated SMTP etc


    ''iMsg.Configuration.Fields.Item("http://schemas.microsoft.com/cdo/co
nfiguration/smtpaccountname") = "My Name"

    ''iMsg.Configuration.Fields.Item("http://schemas.microsoft.com/cdo/co
nfiguration/sendemailaddress") = """MySelf"" <myself@example.com>"

    ''iMsg.Configuration.Fields.Item("http://schemas.microsoft.com/cdo/co
nfiguration/senduserreplyemailaddress") = """Another"" <a-
nother@example.com>"

    ''iMsg.Configuration.Fields.Item("http://schemas.microsoft.com/cdo/co
nfiguration/smtpauthenticate") = cdoBasic

    ''iMsg.Configuration.Fields.Item("http://schemas.microsoft.com/cdo/co
nfiguration/sendusername") = "domain\username"

    ''iMsg.Configuration.Fields.Item("http://schemas.microsoft.com/cdo/co
nfiguration/sendpassword") = "password"

          iMsg.from = EmailReplyToName
          iMsg.to = EmailDstName
          iMsg.subject = "[SUS Server] " & NumNewPatches & " New Patch" &
strPlural & " Approved"
          iMsg.textbody = strMsgBody
          iMsg.send

          LogIt("Sent:" & vbNewline & strMsgBody)

    end if

else
    LogIt("No new patches to approve ")
End if

sub LogIt(logtxt)
    if (debugflag) then
          wscript.echo(now() & ": " & logtxt)
    end if
end sub
```

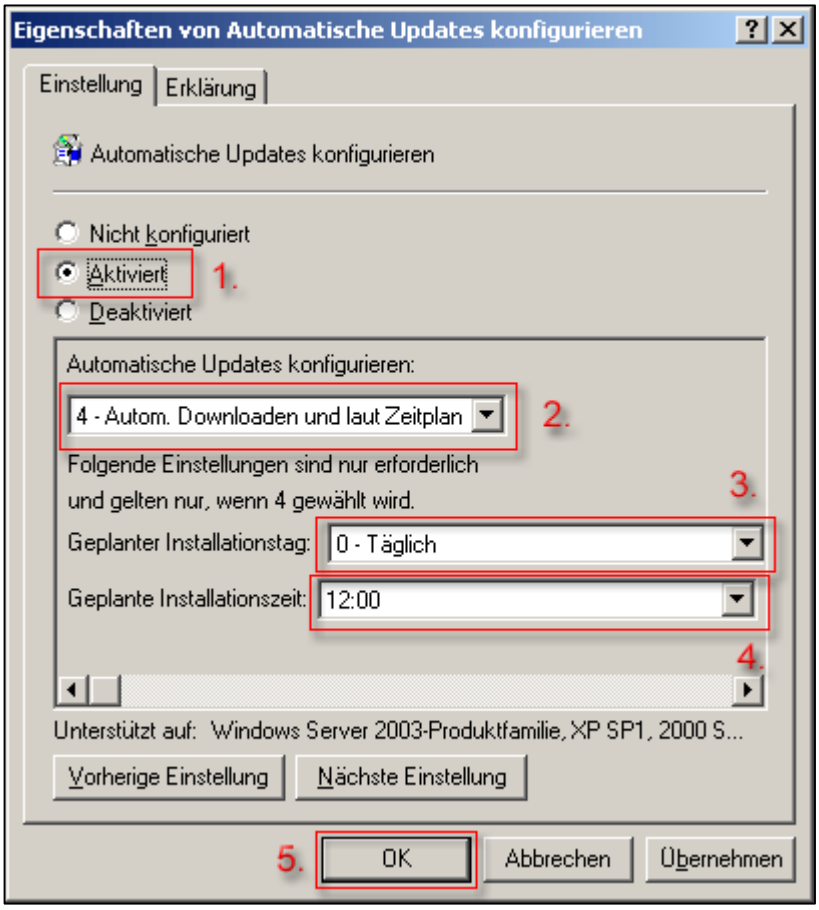
Wie man erkennen kann gibt es in dem Script mehrere kommentierte Zeilen um das Script besser an die eigenen Bedürfnisse anzupassen. Bei `strSUSpath` sollte das Laufwerk mit der Webserver Installation angegeben werden. Wie man erkennen kann ist es sogar möglich eine Remoteverbingung zu einem anderen Server mit der administrativen Freigabe anzugeben. Wichtig ist, dass auf der angegebenen Partition der Webserver installiert wurde. Des Weiteren kann man mit `true` oder `false` entscheiden, welche *Service Packs* eingespielt werden sollen. Mit Sicherheit ist die Option mit `false` bei dem Windows 2003 Server zu setzen, da kein Server unbeaufsichtigt aktualisiert werden sollte. Man

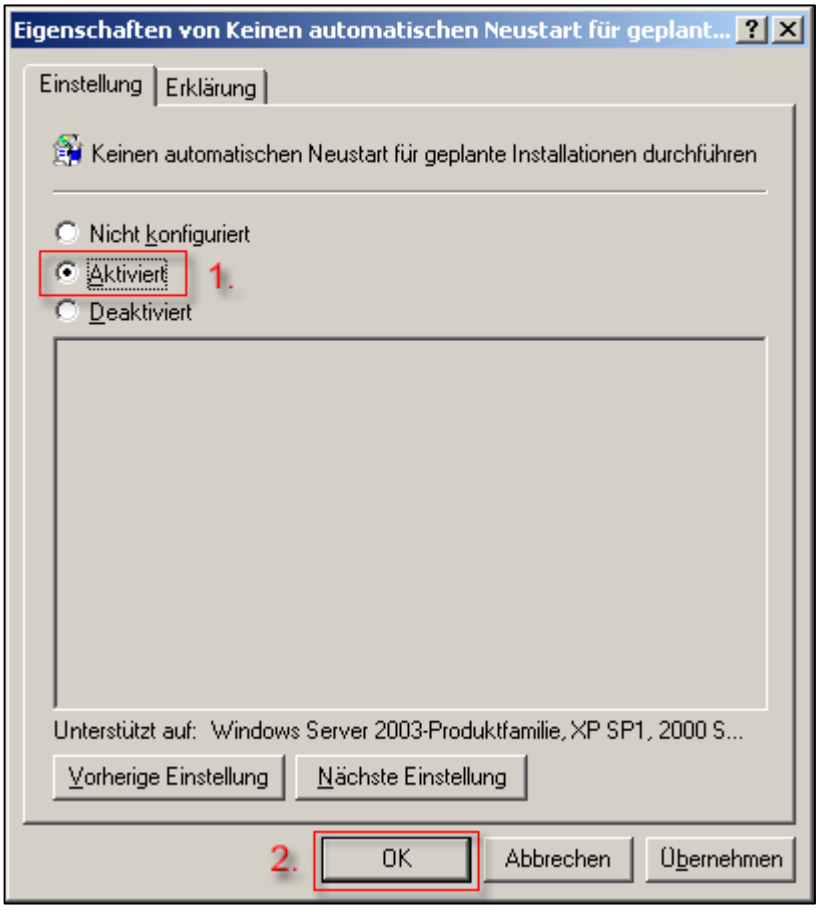
kann als kleines Extra sogar noch die neu freigegebenen Updates sich als Mail zusenden lassen. Hierzu sind wichtig anzugeben Absender- und Empfängeradresse, sowie *SMTP-Server*. Um das Script auszuführen öffnet man die `MS-DOS-Eingabeaufforderung` per `Start` **à** `Ausführen` **à** `cmd`. In der DOS-Box wechselt man auf das Laufwerk und in das Verzeichnis wo sich das Script befindet. Per Eingabe von `cscript dateiname.vbs` startet die Verarbeitung. Um nun alles zu automatisieren legt man für das Script einen neuen *Task* an. Dies geschieht mit dem Aufruf von `Start` **à** `Einstellungen` **à** `Systemsteuerung` **à** `Geplante Tasks` **à** `Geplanten Task hinzufügen`. Jetzt werden alle neu herunter geladenen Updates automatisch freigegeben ohne einen Benutzereingriff.

Das *Visual Basic Script* stammt von der Website www.wsus.info und das Copyright liegt nach wie vor bei www.wsus.info.

Die Arbeitsstationen haben derzeit noch keine Ahnung vom internen SUS-Server. Um diesen bekannt zu geben wird eleganter weise eine Gruppenrichtlinie verwendet. Dazu ist zu sagen, dass für die erfolgreiche Abarbeitung der Gruppenrichtlinie, sich mindestens ein *Microsoft Windows 2000 Professional* mit *Service Pack* 2 auf der Arbeitsstation befinden muss. Die benötigte Gruppenrichtlinie wurde erst mit dem erscheinen von *Microsoft Windows XP* veröffentlicht. Aus diesem Grund hat der *Microsoft Windows 2000 Server* diese nicht in seiner Ausstattung. Die Datei `wuau.adm` kann aber über das im Thema Gruppenrichtlinien angewandte Verfahren eingepflegt werden. Zu laden ist die Datei über die `Computerkonfiguration` des `Gruppenrichtlinien` Snap-In's mit einem Rechtsklick auf die `Administrative Vorlage`. Die gesuchte Rubrik findet man unter `Administrative Vorlagen` **à** `Windows-Komponenten` **à** `Windows Update`. Die Richtlinien `Interner Pfad für den Microsoft Updatedienst angeben`, `Automatische Updates konfigurieren` und `Keinen automatischen Neustart für geplante Installationen durchführen` müssen angepasst werden. Die Einstellungen sollten von den Screenshots übernommen werden.




Eigenschaften von Automatische Updates konfigurieren
Einstellung
Erklärung
Automatische Updates konfigurieren
Nicht konfiguriert
Aktiviert
1.
Deaktiviert
Automatische Updates konfigurieren:
4 - Autom. Downloaden und laut Zeitplan
2.
Folgende Einstellungen sind nur erforderlich
und gelten nur, wenn 4 gewählt wird.
3.
Geplanter Installationstag:
0 - Täglich
Geplante Installationszeit:
12:00
4.
Unterstützt auf: Windows Server 2003-Produktfamilie, XP SP1, 2000 S...
Vorherige Einstellung
Nächste Einstellung
5.
OK
Abbrechen
Übernehmen


Eigenschaften von Keinen automatischen Neustart für geplant...
Einstellung
Erklärung
Keinen automatischen Neustart für geplante Installationen durchführen
Nicht konfiguriert
Aktiviert
1.
Deaktiviert
Unterstützt auf: Windows Server 2003-Produktfamilie, XP SP1, 2000 S...
Vorherige Einstellung
Nächste Einstellung
2.
OK
Abbrechen
Übernehmen