

# PXE

Aus Synology Wiki

Da Preboot Execution Environment (PXE) in der Community ein sehr gefragtes Thema ist, und es meiner Meinung nach immer noch recht schwierig ist, gerade für Einsteiger, vernünftige Anleitungen zu finden, habe ich mich der Sache nun mal angenommen und diesen Wiki-Eintrag verfasst. Gerade mit DSM 4.2 ist das Thema PXE, sofern man weiß wie, doch recht einfach geworden und auch mit keinerlei Linux Kenntnissen umsetzbar geworden. Um es Anderen in Zukunft zu ersparen stundenlang nach irgend etwas Brauchbarem zu suchen werde ich hier auf alles Nötige eingehen.

## Inhaltsverzeichnis

- 1 DSM fit machen für PXE
  - 1.1 DSM 4.2
  - 1.2 DHCP Server einrichten
  - 1.3 Gemeinsamen Ordner für PXE erstellen
  - 1.4 NFS-Dienst aktivieren
  - 1.5 Herunterladen der PXE Konfigurationsdateien
  - 1.6 TFTP/PXE Server einrichten
  - 1.7 Testen ob alles geklappt hat
- 2 Die PXE Konfigurationsdatei "default"
- 3 Images
  - 3.1 Acronis Disc Director 10
  - 3.2 CloneZilla
    - 3.2.1 Speichern/Wiederherstellen eines Backups auf die DiskStation
  - 3.3 Kali Live
  - 3.4 Kali (netboot)
  - 3.5 Memtest86+
  - 3.6 Partition Wizard Home Edition
  - 3.7 Ultimate Boot CD
  - 3.8 Ubuntu Live
  - 3.9 Ubuntu (netboot)
  - 3.10 Win7PE / Win8PE
  - 3.11 Windows 7 / 8 / 8.1 / 10
    - 3.11.1 Erstellen einer eigenen WinPE.iso mit verändertem Startskript
    - 3.11.2 Erstellen einer eigenen WinPE.iso mit verändertem Startskript (Windows 10 only)

## DSM fit machen für PXE

### DSM 4.2

Wie schon erwähnt setzte ich **DSM**  $\geq$  **4.2** voraus, da man mit dieser Version nun seine DiskStation als TFTP Server einsetzen und PXE über die Oberfläche konfigurieren kann.

### DHCP Server einrichten

Als erstes müssen wir unsere DiskStation zu einem DHCP-Server machen. Dies ist nötig, da nur ein DHCP-Server spezielle, für PXE nötige Einstellungen den Clients übertragen kann. Da in den meisten Fällen der handelsübliche Router zu Hause dazu nicht in der Lage ist, muss das ab sofort unsere DiskStation nun übernehmen.

Dazu installieren wir uns (falls noch nicht geschehen) das **"DHCP Server"-Paket aus dem Paketzentrum**, starten und öffnen es. Bei DiskStations mit mehr als nur einer LAN-Schnittstelle öffnet sich nun nicht wie vielleicht erwartet ein Fenster "DHCP Server", sondern zunächst die Systemsteuerung im Unterpunkt Netzwerk, mit dem geöffneten Reiter "Netzwerk-Schnittstelle". Hier wählen wir nun die Schnittstelle aus, welche am Ende DHCP-Adressen vergeben soll und drücken dort dann auf "DHCP Server". Unter dem Reiter "Allgemein" aktivieren wir zunächst einmal das Kästchen **"DHCP Server aktivieren"**, stellen eine gewünschte Lease-Time ein (360 Minuten z.B.) und tragen unseren Primären DNS-Server ein (das ist in der Regel der Router, dieser hat bei mir die 192.168.178.1). In der Teilnetz-Liste fügen wir einen neuen Bereich hinzu. Dieser dient lediglich dafür die Funktionalität des vorhandenen DHCP-Servers vom Router zu ersetzen. Wer sich dieser Angaben unsicher ist schaut in der Konfiguration seines Routers nach und schreibt diese dort ab. Meine

sehen hier z.B. so aus:

- Start-IP-Adresse: 192.168.178.100
- End-IP-Adresse: 192.168.178.200
- Netmask: 255.255.255.0
- Gateway: 192.168.178.1

Als Letztes muss noch der **DHCP-Server vom Router**, der in den meisten Fällen standardmäßig an ist, **ausgeschalten** werden. Dies ist äußerst Wichtig, da dieser, wie schon erwähnt, später nicht die nötigen PXE-Informationen an die Clients verteilt und wir nur so sicher gehen können, das die Clients auch den richtigen DHCP-Server, nämlich die DiskStation, verwenden!

## Gemeinsamen Ordner für PXE erstellen

Für alle weiteren Schritte benötigen wir einen neuen Gemeinsamen Ordner für unsere PXE Konfigurationsdateien und später für die Images. Bei mir liegt dieser unter **/volume1/PXE**

Sorgt in den Privilegieneinstellungen für Folgendes: **admin darf Lesen/Schreiben** und der **guest darf Lesen** ... den Rest könnt Ihr auf Keinen Zugriff bzw. maximal auf Lesen stellen. Schließlich soll hier keiner außer der admin an der Konfiguration rumwerkeln dürfen!

## NFS-Dienst aktivieren

Für die meisten UNIX und LINUX PXE-Images werden NFS-Berechtigungen benötigt, deshalb greife ich an dieser Stelle schonmal vor und wir stellen diese nun hier ein. Also öffnen wir bei unserem Gemeinsamen Ordner unter Berechtigungen die NFS-Berechtigungen und klicken auf Erstellen:

1. Unter Hostname oder IP\* geben wir die IP-Adressen an, die später die Images nutzen dürfen. Ich mache hier keinerlei Einschränkungen und sage das ganze Netz darf zugreifen, also trage ich folgendes ein: **192.168.178.0/24**
2. Bei Privileg sage ich wiederum: **Nur Lesen**
3. Root squash lassen wir auf: **Keine Zuordnung**
4. Sicherheit lassen wir ebenfalls auf: **sys**
5. Häkchen bei: **Asynchron aktivieren**
6. Verbindungen von nicht-privilegierten Ports zulassen lassen wir **aus**
7. 2 mal OK und das war es mit dem Gemeinsamen Ordner
8. Als letztes vergessen wir nicht auch den **NFS-Dienst** zu **aktivieren**. Dies geht unter Systemsteuerung / "Win/Mac/NFS" / NFS-Dienst. Ein einfaches setzen des Häkchens genügt.

## Herunterladen der PXE Konfigurationsdateien

Auf Github gibt es eine Sammlung der wichtigsten PXE Konfigurationsdateien. Diese laden wir uns, entpacken und kopieren sie in unseren Gemeinsamen Ordner PXE: TFTP-PXE-Boot-Server (<https://github.com/paulmaunders/TFTP-PXE-Boot-Server>) (Rechts unten gibt es den Button "Download Zip")

**Vorsicht!:** Die Zip enthält einen Ordner mit den Dateien. Wir kopieren NICHT den Ordner in unseren Gemeinsamen Ordner, sondern lediglich dessen Inhalt, sodass letztendlich die Konfigurationsdateien wie z.B. "pxelinux.0" direkt in unserem Gemeinsamen Ordner "PXE" liegen! (Die Ordner "pxelinux.cfg" und "images" bleiben jedoch erhalten!)

## TFTP/PXE Server einrichten

1. Systemsteuerung / FTP -> Reiter TFTP/PXE in der DSM öffnen
2. Häkchen bei: **TFTP-Server aktivieren**
3. Als "**TFTP-Root Ordner**" den oben erstellten Gemeinsamen Ordner "**PXE**" auswählen. In anderen Tutorials im Internet wird dieser Ordner oft nur "**ftpproot**" genannt.
4. Unter "Erweiterte Einstellungen" erlauben wir allen Clients wieder nur Lesen und lassen das Zeitlimit auf 3 stehen. Im Protokoll kann man letztendlich sehen welcher Client, welche Datei gezogen hat. Wer das braucht kann dies aktivieren
5. Nun aktivieren wir das Kästchen "**DHCP-Dienst auf diesem Server für PXE einrichten**" und machen folgende Einstellungen:
  - **Bootloader: pxelinux.0** (Falls diese Datei hier nicht sichtbar ist, haben wir etwas beim "Herunterladen der PXE Konfigurationsdateien" falsch gemacht)
  - Netzwerkschnittstelle: LAN 1 (Hier wählen wir die gleiche Schnittstelle wie oben beim DHCP-Server!)

- DNS-Server: 192.168.178.1 (wieder genau wie oben beim DHCP-Server)
- Start-IP-Adresse: 192.168.178.201 (Sollte eigentlich kein Problem sein, wenn sich der oben angegebene Bereich mit diesem hier überschneidet. Weil es aber so schöner aussieht und dieser Bereich nun auch in den DHCP-Einstellungen sichtbar ist, solltet Ihr das auch nicht überschneidend angeben)
- End-IP-Adresse: 192.168.178.250
- Netzmaske: 255.255.255.0
- Gateway: 192.168.178.1 (wieder der Router)

6. Übernehmen und fertig!

**Hinweis:** Der hier angegebene IP-Adress-Bereich unterscheidet sich NICHT wie eventuell vermutet in der Funktionsweise von dem im DHCP-Server. Beide IP-Bereiche geben letztlich die nötigen PXE Informationen an die Clients mit, ob benötigt oder nicht. Der hier angegebene IP-Adress-Bereich ist aber nicht Optional und muss angegeben werden. Falls man also nur einen Bereich haben möchte, so kann man den im DHCP-Server weglassen und nur diesen hier angeben.

## Testen ob alles geklappt hat

Haben wir alle Einstellungen richtig gemacht sollte unser PXE-Server nun laufen und einsatzbereit sein. Um das ganze zu Testen können wir nun entweder einen PC/Laptop als Client benutzen oder z.B. eine VirtualBox dafür nehmen.

- Einstellungen für den PC oder Laptop:
  1. Neustarten und in das jeweilige BIOS des Rechners gehen
  2. Dafür sorgen, dass das **Netzwerk in der Bootreihenfolge an erster Stelle** steht
  3. Optional: Bei manchen BIOS muss man das Netzwerk Boot Protocol noch separat auswählen. Dieses stellt man dann logischerweise auf PXE (Preboot Execution Environment)

Auf jedes einzelne BIOS kann und werde ich an dieser Stelle nicht eingehen. Die wichtigsten Punkte wurden hier genannt.

- Einstellungen für die VirtualBox:
  1. Erstellt euch eine neue Virtuelle Maschine. Name, Betriebssystem, Version, Speicher- sowie Festplattengröße ist hier egal. Ihr könnt auch eine bereits erstellte Virtuelle Maschine verwenden.
  2. Geht in die Einstellungen (Ändern), dort auf "Netzwerk" und stellt den Adapter, der ans Netzwerk angeschlossen ist auf **"Netzwerkbrücke"** um. Sonst ist dieser Adapter nicht im "echten" Netzwerk und kann keine DHCP-Informationen von unserer DiskStation erhalten
  3. Wir starten nun unsere Virtuelle Maschine und drücken ziemlich flott auf **"F12"** um das Booten zu unterbrechen
  4. Jetzt drücken wir **"L"**, um vom LAN zu Booten
  5. Um permanent als erstes vom Netzwerk zu Booten können wir das in den Einstellungen auch fest einstellen. Hierzu gehen wir in die Einstellungen (Ändern), dort auf "System/Hauptplatine" und stellen "Netzwerk" in der Bootreihenfolge an erste Stelle und bestätigen mit "OK"

Hat mit unserer PXE-Konfiguration nun alles geklappt sollten wir eine DHCP-Adresse von unserer DiskStation erhalten und kurze Zeit später unser blauer PXE-Auswahlbildschirm erscheinen.

## Die PXE Konfigurationsdatei "default"

Hier werde ich kurz auf die Konfigurationsdatei "default" im Ordner "pxelinux.cfg" eingehen. So (ausgenommen vom Eintrag MENU TITLE) sieht sie aus nachdem wir sie geladen haben:

```

-----
default menu.c32
prompt 0
timeout 100
ONTIMEOUT chainlocal
MENU TITLE PXE Boot Menu - Synology DiskStation

LABEL local
    MENU LABEL Boot local hard drive
    LOCALBOOT 0

LABEL chainlocal
    MENU LABEL Chain boot to local hard drive
    KERNEL chain.c32
    APPEND hd0

LABEL CentOS-6.3-x86_64-eth0-gpt
    MENU LABEL Install CentOS 6.3 x86_64 eth0 - GPT version for 2TB+ disks
    KERNEL images/centos/6.3/x86_64/vmlinuz
    APPEND ks=http://fubralimited.github.io/CentOS-KVM-Image-Tools/kickstarts/centos6x-hypervisor-gpt-selinux.cfg initrd=images/centos/6
    LABEL CentOS-6.3-x86_64-eth0-mbr
  
```

```
MENU LABEL Install CentOS 6.3 x86_64 eth0 - MBR version for older BIOS
KERNEL images/centos/6.3/x86_64/vmlinuz
```

- **default menu.c32** legt hier den Standard Menü-Stil fest. "menu.c32" ist hier der Textbasierte Stil mit blauem Hintergrund. Alternative hierfür wäre der grafische Menü-Stil "vesamenu.c32". Da hierfür mehrere Schritte vonnöten sind und es mir hier auf das Wesentliche ankommt, werde ich hier auf den grafischen Menü-Stil nicht weiter eingehen.
- **prompt 0** bedeutet, dass der Fokus beim Laden des PXE Bootscreens auf dem ersten Eintrag liegt. (0 = 1. Eintrag, 1 = 2. Eintrag ... usw.)
- **timeout 100** bedeutet, dass nach 10,0 Sekunden ohne Interaktion durch den Benutzer automatisch der Eintrag ausgewählt wird, der bei **ONTIMEOUT** hinterlegt wurde. In diesem Fall "chainlocal".
- **MENU TITLE** legt die Überschrift des PXE Bootmenüs fest, hier: "PXE Boot Menu - Synology DiskStation"

Nun folgen die einzelnen Menüeinträge, deren Reihenfolge nach Belieben geändert werden kann. Ein Eintrag ist wie folgt gegliedert:

- **LABEL** ist hier einfach nur der Name des Eintrags in der Konfigurationsdatei selber. Dieser muss aus einem Wort bestehen (ohne Leer- oder Sonderzeichen), ist total irrelevant und kommt auch nirgendwo sonst vor und dient nur zur Orientierung in der Konfigurationsdatei. Alle weiteren Punkte sind Unterpunkte von LABEL und sind lediglich zur Optik eingerückt (kein Muss!)
  - **MENU LABEL** Hier wird der Titel des Eintrags im PXE-Boot-Menü festgelegt und kann frei gewählt werden.
  - **KERNEL** Hier wird der Pfad zum verwendeten (Linux-)Kernel angegeben. Bei dem Pfad wird hier immer vom oben ausgewählten "TFTP Root-Ordner" ausgegangen. Später liegen (mit Ausnahmen) alle Kernel in "images/..."
  - **APPEND** Hier können bzw. werden für die einzelnen Kernel gewisse Parameter mitgegeben. Diese unterscheiden sich von Kernel zu Kernel. Diese Zeile kann bei einigen Images durch das Angeben vieler Parameter schon mal ziemlich lang werden. Wir achten hierbei darauf, diese Zeile nicht etwa durch einen Zeilenumbruch zu unterbrechen!
  - **LOCALBOOT 0** ist der Eintrag, der immer als Erstes stehen sollte und ganz einfach die erste lokale Festplatte bootet.

Die Reihenfolge der Menüeinträge ist egal und kann frei gewählt werden. Weitere Einträge müssen einfach der Datei angehängt werden. Um die Konfigurationsdatei übersichtlicher zu gestalten habe ich die letzten drei Standard Einträge, chainlocal und die beiden CentOS-Einträge in den folgenden Image-Tutorials gelöscht.

**Hinweis:** Eine vollständige Dokumentation über die Einträge, die in der default möglich sind gibt es hier: [syslinux.org - menu.c32](http://www.syslinux.org/wiki/index.php/Comboot/menu.c32) (<http://www.syslinux.org/wiki/index.php/Comboot/menu.c32>) Hier gibt es ganz nützliche Dinge wie Untermenüs, Trenner, Farben, Passwörter usw. um das PXE-Boot-Menü etwas übersichtlicher und schöner zu gestalten!

## Images

- Für alle Images setze ich voraus, dass "DSM fit machen für PXE" durchgeführt wurde und läuft
- Die geposteten Konfigurationen sind nur die jeweiligen Menüeinträge der default PXE-Konfigurationsdatei und müssen einfach nur angehängt werden ( pxelinux.cfg/default )
- Bei allen relativen Pfadangaben gehe ich von unserem Gemeinsamen Ordner PXE aus ( /volume1/PXE )

## Acronis Disc Director 10

Acronis Disc Director ist ein Partitions-Manager, um z.B. unsere Windows Partition auch im Nachhinein noch zuverlässig vergrößern oder verkleinern zu können.

1. Erstellen eines neuen Ordners "acronis" in "PXE/images/"
2. Jetzt gibt es zwei Möglichkeiten. Ihr bezieht die ramdisk und den kernel von einer vorhandenen CD oder von der bereits installierten Version von Acronis Disc Director 10 aus dem Installationsverzeichnis:
  - CD-Version:
    1. Einlegen und öffnen der CD
    2. Kopieren der Dateien "DAT2.DAT" und "DAT3.DAT" in unseren Ordner "PXE/images/acronis"
    3. Umbenennen der kopierten Datei "DAT2.DAT" in "ramdisk.dat"
    4. Umbenennen der kopierten Datei "DAT3.DAT" in "kernel.dat"
  - Installationsverzeichnis:
    1. Kopieren der "kernel.dat" und "ramdisk.dat" in unseren Ordner "PXE/images/acronis"
3. Als Letztes fügen wir noch den folgenden Menüeintrag an das Ende unserer default PXE-Konfigurationsdatei an:

```
LABEL acronis
MENU LABEL Acronis Disc Director 10
KERNEL images/acronis/kernel.dat
APPEND initrd=images/acronis/ramdisk.dat vga=791 ramdisk_size=102400 quiet
```

- Unter Umständen kann es vorkommen, dass die Maus zwecks fehlenden Treibern nicht bei jedem funktioniert. So muss man sich halt mit der Tastatur durchs Menü kämpfen.

## CloneZilla

Mit CloneZilla lassen sich komplette Partitionen oder Festplatten auf unserer DiskStation sichern und wiederherstellen.

1. Erstellen eines neuen Ordners "clonezilla" in "PXE/images/"
2. Downloaden der aktuellsten CloneZilla Version von: [clonezilla.org](http://clonezilla.org/downloads.php) (<http://clonezilla.org/downloads.php>). Ob ihr das stable oder testing release nehmt müsst Ihr entscheiden. Achtet auf die richtige Architektur (ich hab die amd64 Version genommen) und zieht die zip-Version, nicht die iso-Version.
3. Entpackt euer Archiv und kopiert nur den Inhalt vom Ordner "live" in euren erstellten Ordner "PXE/images/clonezilla"
4. Als Letztes fügen wir noch den folgenden Menüeintrag an das Ende unserer default PXE-Konfigurationsdatei an:

```
-----
LABEL clonezilla
MENU LABEL CloneZilla Live 2.1.2-43 x64
KERNEL images/clonezilla/vmlinuz
APPEND initrd=images/clonezilla/initrd.img boot=live config noswap nolocales edd=on nomodeset ocs_live_run="ocs-live-general" ocs_li
```

- Vergesst nicht die IP-Adresse eurer DiskStation anstatt der meiner beim tftp-Pfad einzutragen!
- Wer sich über den tftp-Pfad wundert und meint einen Schreibfehler von mir gefunden zu haben, da hier das "PXE" im Pfad nach der IP-Adresse fehlt, irrt sich. Habe mich darüber auch erst gewundert, aber er sucht hier bei der angegebenen IP-Adresse direkt im tftpboot. Es funktioniert auch nur so und dabei belasse ich es auch.
- Weitere Parameter, die Ihr beim Booten von CloneZilla mitgeben könnt, findet Ihr **HIER** ([http://clonezilla.org/show-live-doc-content.php?topic=clonezilla-live/doc/99\\_Misc](http://clonezilla.org/show-live-doc-content.php?topic=clonezilla-live/doc/99_Misc))

## Speichern/Wiederherstellen eines Backups auf die DiskStation

1. Als erstes sollten wir sicherstellen, dass wir für unser Backup auf die DiskStation einen Gemeinsamen Ordner und einen Benutzer mit Schreibrechten haben. **Wichtig:** Wir benutzen NICHT unseren Gemeinsamen Ordner "PXE" für dieses Vorhaben, obwohl vielleicht der Ordner "images" vom Namen her dazu verleiten könnte! Ich habe mich hier für einen neuen Order "Backups" (/volume1/Backups) entschieden und einen neuen Benutzer "pxeuser", der darauf Schreibrechte hat.
2. Wir Booten CloneZilla und wählen bei der ersten Aufforderung "**Starte CloneZilla**"
3. Jetzt "**device-image**", da wir ja auf keine Partition auf der DiskStation speichern wollen, sondern in ein Image
4. Jetzt "**samba-server**" und schreiben die IP unserer DiskStation rein, bei mir: **192.168.178.250**
5. Bei Domain geben wir "**Cancel**" ein
6. Jetzt folgt der Benutzer, der Schreibrechte für den Gemeinsamen Ordner hat. Bei mir: **pxeuser**
7. Jetzt erfolgt der Gemeinsame Ordner für unsere Backups, dieser beginnt immer mit einem "/": **/Backups**
8. Sicherheitsverfahren: **auto Systemstandard verwenden**
9. Jetzt wird man aufgefordert das Passwort für den Benutzer einzugeben: **pxeuser\_password**
10. Wir bestätigen mit "**ENTER**"
11. Beim Assistenten wählen wir "**Beginner**"
12. Jetzt können wir uns zwischen mehreren Optionen entscheiden:
  - savedisk - Lokale Platte als Image sichern
  - saveparts - Lokale Partition(en) als Image sichern
  - restoredisk - Image auf Lokale Platte wiederherstellen
  - restoreparts - Image auf Lokale Partition wiederherstellen
  - ...
13. Wir entscheiden uns für eine gewünschte Option und folgen dem Assistenten. Ab hier sollte alles weitere klar sein.

## Kali Live

Kali ist der offizielle Nachfolger von BackTrack 5. Mit dieser Methode lässt sich ein vorhandenes Kali-ISO direkt von der DiskStation starten.

1. Erstellen eines neuen Ordners "kali" in "PXE/images/"
2. Downloaden eines von euch gewünschten ISOs eines Kali-Releases von [kali.org](http://www.kali.org/downloads/) (<http://www.kali.org/downloads/>) (Ich nehme hier als Beispiel: Kali 1.0.5 x64)
3. Entpacken der ISO in euren erstellten Ordner "PXE/images/kali" (**nur für OS X Benutzer:** das Öffnen des ISOs mit dem Finder kann die Fehlermeldung "Keine aktivierbaren Dateisysteme" auslösen. Die Lösung hierfür wäre ein manuelles mounten der ISO mit dem Terminal. Damit Ihr euch aber nicht unnötig mit Kommandozeilenbefehle auseinandersetzen müsst, empfehle ich euch das Programm Keka (<http://www.kekaosx.com/de/>) herunterzuladen. Das ist ein kostenloses Archivierungsprogramm für OS X, dass die ISO problemlos entpacken kann)
  - **Hinweis nur für Linux und OS X Benutzer:** Wer das ISO lokal entpackt und erst anschließend auf seine DiskStation kopiert

hat, sollte darauf achten auch den versteckten Ordner ".disk" mitkopiert zu haben! Das Booten des ISOs ist zwar im Gegensatz zum Ubuntu-ISO auch ohne diesen Ordner fehlerfrei möglich, jedoch lege ich hier einfach Wert auf Vollständigkeit.

- Die vorhandene Ramdisk "live/initrd.img" in eurem Ordner "kali" ist zum jetzigen Zeitpunkt noch nicht in der Lage auf die DiskStation via NFS zuzugreifen und wird Fehler bringen, da entsprechende Module fehlen. Diese müssen per Hand/Konsole aus einem gebooteten Kali kopiert und in die initrd.img eingefügt werden. Da diese Prozedur mal wieder ein bisschen Konsolenarbeit erfordert habe ich das für euch schonmal gemacht. Ladet euch die entsprechende initrd.img hier runter und ersetzt sie mit eurer in "PXE/images/kali/live/":
  - initrd.img x86 (ca. 21MB) (<http://ul.to/shboaoh>)
  - initrd.img x64 (ca. 17MB) (<http://ul.to/0mpro7kv>)
  - Für alle die wissen wollen wie das mit den Modulen funktioniert hat, hier die Anleitung (<https://wiki.hashdump.org/index.php/PXE::Kali>)
- Als Letztes fügen wir noch den folgenden Menüeintrag an das Ende unserer default PXE-Konfigurationsdatei an:

```

LABEL kali
  MENU LABEL Kali 1.0.5 x64 - Live
  KERNEL images/kali/live/vmlinuz
  APPEND noconfig=sudo username=root hostname=kali root=/dev/nfs boot=live netboot=nfs nfsroot=192.168.178.250:/volume1/PXE/images/kali

```

- Vergesst nicht die IP-Adresse eurer DiskStation anstatt der meiner beim nfsroot und tftp-Pfad einzutragen!
- Auch der nfsroot-Pfad könnte geändert werden müssen. Sollte euer Gemeinsamer Ordner z.B. auf der zweiten Festplatte in eurer DiskStation liegen, so sollte hier jetzt /volume2/PXE/... stehen.
- Wer jetzt beim Booten noch so eine Fehlermeldung bekommt:

```

mount call failed - server replied: Permission denied

```

der sollte sich das Kapitel 1.4 noch mal genauer anschauen bzw. seine NFS-Rechte überprüfen!

## Kali (netboot)

Kali ist der offizielle Nachfolger von BackTrack 5. Mit dieser Methode können wir das aktuellste Kali-Release direkt aus dem Internet laden und installieren.

- Downloaden folgender Datei (ca. 15MB):

```

http://repo.kali.org/kali/dists/kali/main/installer-amd64/current/images/netboot/netboot.tar.gz

```

Die Architektur kann von "amd64" in "i386" geändert werden.

- Wir entpacken das Archiv und kopieren aus dem Ordner "netboot.tar" den Ordner "debian-installer" in unseren Gemeinsamen Ordner PXE. (**Wichtig:** Der Ordner MUSS in PXE liegen und NICHT wie bei anderen Images in PXE/images, weil hier ein anderer bootloader verwendet wird und dieser auf genau diesen Pfad angepasst wurde!)
- Als Letztes fügen wir noch den folgenden Menüeintrag an das Ende unserer default PXE-Konfigurationsdatei an:

```

LABEL kali_netboot
  MENU LABEL Kali installieren x64 (Internet)
  KERNEL debian-installer/amd64/boot-screens/vesamenu.c32
  APPEND debian-installer/amd64/boot-screens/menu.cfg

```

- **Vorsicht:** Wer anstatt der amd64 Version die i386 Version geladen hat, muss dies auch im Menüeintrag abändern!

## Memtest86+

Memtest86+ ist ein hervorragendes Programm, um unseren Arbeitsspeicher (RAM) auf Fehler zu überprüfen.

- Erstellen eines neuen Ordners "memtest" in "PXE/images/"
- Downloaden der aktuellsten Memtest86+ Version von: [memtest.org](http://memtest.org/#downiso) (<http://memtest.org/#downiso>) (Pre-Compiled Bootable Binary (.zip) )
- Entpacken des Archivs und kopieren der "memtest86+-5.01.bin" in unseren Ordner "PXE/images/memtest"
- Umbenennen der kopierten Datei "memtest86+-5.01.bin" in "memtest86+" (**Wichtig:** Achtet darauf auch die Dateiendung zu entfernen!)
- Als Letztes fügen wir noch den folgenden Menüeintrag an das Ende unserer default PXE-Konfigurationsdatei an:

```

LABEL memtest86

```

```
MENU LABEL Memtest86+ 5.01
KERNEL images/memtest/memtest86+
```

- Sollte beim Auswählen von Memtest86+ im PXE-Boot-Menü einfach nichts passieren, so ist es höchst wahrscheinlich, dass Ihr vergessen habt die Dateieindung (.bin) mit zu löschen!

## Partition Wizard Home Edition

Partition Wizard Home Edition ist ähnlich wie Acronis Disc Director ein Partitions-Manager, um z.B. unsere Windows Partition auch im Nachhinein noch zuverlässig vergrößern oder verkleinern zu können. Anders als Acronis Disc Director, unterstützt Partition Wizard auch die dynamischen Partitionen von Windows 8 und 8.1.

1. Erstellen eines neuen Ordners "partitionwizard" in "PXE/images/"
2. Downloaden des aktuellsten Partition Wizard ISOs von: [partitionwizard.com](http://www.partitionwizard.com/download.html) (Free Download Bootable CD ganz unten, ca. 50MB) (<http://www.partitionwizard.com/download.html>)
3. Das ISO mounten und im Ordner "BOOT" alle Dateien in unseren Ordner "PXE/images/partitionwizard" kopieren
4. Als Letztes fügen wir noch den folgenden Menüeintrag an das Ende unserer default PXE-Konfigurationsdatei an:

```
LABEL partitionwizard
MENU LABEL Partition Wizard Home Edition
KERNEL images/partitionwizard/BZIMAGE
APPEND initrd=images/partitionwizard/tinycore.gz ramdisk_size=102400 root=/dev/ram0 rw
```

## Ultimate Boot CD

Mit dieser Methode lässt sich das aktuellste UBCD ISO direkt von der DiskStation starten. Die Ultimate Boot CD (UBCD) ist eine komplette und umfangreiche Sammlung von System-Utilities auf einer bootfähigen CD.

1. Erstellen eines neuen Ordners "ubcd" in "PXE/images/"
2. Downloaden der aktuellsten UBCD ISO von: [ultimatebootcd.com](http://www.ultimatebootcd.com/download.html) (<http://www.ultimatebootcd.com/download.html>) (ISO Image, ca. 575MB)
3. Kopieren der heruntergeladenen "ubcd529.iso" in unseren Ordner "PXE/images/ubcd"
4. Als Letztes fügen wir noch den folgenden Menüeintrag an das Ende unserer default PXE-Konfigurationsdatei an:

```
LABEL ubcd
MENU LABEL Ultimate Boot CD 5.2.9
KERNEL memdisk
APPEND iso initrd=images/ubcd/ubcd529.iso
```

- Bei neueren Versionen muss natürlich der Name und Pfad entsprechend geändert werden

## Ubuntu Live

Mit dieser Methode lässt sich das von euch gewählte (heruntergeladene) Ubuntu-ISO direkt von der DiskStation starten und über ein Desktop-Shortcut dann auch installieren.

1. Erstellen eines neuen Ordners "ubuntu" in "PXE/images/"
2. Downloaden eines von euch gewünschten ISOs eines Ubuntu-Releases. (Ich nehme hier als Beispiel: Ubuntu 12.04.02 LTS x64 CD - Live)
3. Entpacken der ISO in euren erstellten Ordner "PXE/images/ubuntu" (**nur für OS X Benutzer:** das Öffnen des ISOs mit dem Finder kann die Fehlermeldung "Keine aktivierbaren Dateisysteme" auslösen. Die Lösung hierfür wäre ein manuelles mounten der ISO mit dem Terminal. Damit Ihr euch aber nicht unnötig mit Kommandozeilenbefehle auseinandersetzen müsst, empfehle ich euch das Programm Keka (<http://www.kekaosx.com/de/>) herunterzuladen. Das ist ein kostenloses Archivierungsprogramm für OS X, dass die ISO problemlos entpacken kann)
  - **Hinweis nur für Linux und OS X Benutzer:** Wer das ISO lokal entpackt und erst anschließend auf seine DiskStation kopiert hat, sollte darauf achten auch den versteckten Ordner ".disk" mitkopiert zu haben! Dieser ist unerlässlich für das fehlerfreie Booten des Ubuntu-Releases später und wird, falls nicht vorhanden, folgende Fehlermeldung verursachen:

```
Please provide a name for this Disc, such as 'Debian 5.0.3 Disk 1':
```

4. Als Letztes fügen wir noch den folgenden Menüeintrag an das Ende unserer default PXE-Konfigurationsdatei an:

```

LABEL ubuntu
MENU LABEL Ubuntu 12.04.02 LTS x64 CD - Live
KERNEL images/ubuntu/casper/vmlinuz
APPEND boot=casper vga=normal netboot=nfs nfsroot=192.168.178.250:/volume1/PXE/images/ubuntu fetch=tftp://192.168.178.250/images/ubu

```

- Vergesst nicht die IP-Adresse eurer DiskStation anstatt der meiner beim nfsroot und tftp-Pfad einzutragen!
- Auch der nfsroot-Pfad könnte geändert werden müssen. Sollte euer Gemeinsamer Ordner z.B. auf der zweiten Festplatte in eurer DiskStation liegen, so sollte hier jetzt /volume2/PXE/... stehen.
- Das selbe wie bei CloneZilla. Auch hier fehlt NICHT das "PXE" im tftp-Pfad! Er sucht auch hier direkt im tftproot.
- **Vorsicht:** Ich kann generell nicht für die Richtigkeit der Kernel- und Ramdisk-Namen garantieren. Bei z.B. dem gleichen Release (Ubuntu 12.04.02 LTS x64), lediglich als DVD-Version geladen, hieß der Kernel anstatt "vmlinuz" nun "vmlinuz.efi" und die Ramdisk anstatt "initrd.lz" nun "initrd.gz". Diese beiden Namen können von Release zu Release variieren und es liegt an euch nachzusehen und diese dann in der default entsprechend anzupassen!
- Wer jetzt beim Booten noch so eine Fehlermeldung bekommt:

```
mount call failed - server replied: Permission denied
```

der sollte sich das Kapitel 1.4 noch mal genauer anschauen bzw. seine NFS-Rechte überprüfen!

## Ubuntu (netboot)

Mit dieser Methode können wir das aktuellste (angegebene) Ubuntu-Release direkt aus dem Internet laden und installieren.

1. Downloaden folgender Datei (ca. 25MB):

```
http://archive.ubuntu.com/ubuntu/dists/saucy/main/installer-amd64/current/images/netboot/netboot.tar.gz
```

(Wir achten hier im Pfad auf das Wort "**saucy**". Das ist das Release welches wir später installieren. Wer ein anderes Release wünscht, ersetzt das Wort "saucy" mit seinem gewünschten Release. Ebenso kann die Architektur "**amd64**" in "**i386**" geändert werden)

2. Wir entpacken das Archiv und kopieren aus dem Ordner "netboot.tar" den Ordner "ubuntu-installer" in unseren Gemeinsamen Ordner PXE. (**Wichtig:** Der Ordner MUSS in PXE liegen und NICHT wie bei anderen Images in PXE/images, weil hier ein anderer bootloader verwendet wird und dieser auf genau diesen Pfad angepasst wurde!)
3. Als Letztes fügen wir noch den folgenden Menüeintrag an das Ende unserer default PXE-Konfigurationsdatei an:

```

LABEL ubuntu_netboot
MENU LABEL Ubuntu (saucy) installieren x64 (Internet)
KERNEL ubuntu-installer/amd64/boot-screens/vesamenu.c32
APPEND ubuntu-installer/amd64/boot-screens/menu.cfg

```

- **Vorsicht:** Wer anstatt der amd64 Version die i386 Version geladen hat, muss dies auch im Menüeintrag abändern!

## Win7PE / Win8PE

Win7PE bzw. Win8PE ist ein Windows Live System basierend auf Windows 7 bzw. 8. Nützlich, wenn gerade kein Windows auf dem Rechner ist und man eben mal eine Festplatte durchsuchen oder etwas kopieren möchte. Ich erkläre hier alle Schritte für Win7PE. Für Win8PE könnt Ihr exakt das selbe durchführen, Ihr solltet lediglich die "7"er durch "8"er ersetzen ...

1. Erstellen eines neuen Ordners "win7pe" in "PXE/images/"
2. Erstellen eines neuen Ordners "C:\ISO"
3. Kopieren aller Dateien von einer Windows 7 ISO oder DVD in den Ordner C:\ISO
4. Erstellen eines neuen Ordners "C:\WIN7PE" (**Wichtig:** Dieser Ordner MUSS hier liegen, da das Programm sonst einen Fehler verursacht!)
5. Win7PE SE bzw. Win8PE SE herunterladen:
  - Win7PE SE (<http://w7pese.cwcodes.net/Compressed/index.php>) herunterladen (Complete ca. 175MB)
  - Win8PE SE (<http://w8pese.cwcodes.net/Compressed/index.php>) herunterladen (Complete ca. 165MB)
6. Entpacken des Archivs in unseren Ordner "C:\WIN7PE"
7. Nun starten wir Win7PE SE mit "C:\WIN7PE\Win7PESE82\_Builder.exe"
8. Wir klicken auf "Source" und wählen bei "Source directory" unseren Ordner "C:\ISO" mit den enthaltenen Windows 7 Dateien
9. Nun klicken wir auf den blauen Button oben rechts "PLAY" und warten bis das ISO erstellt wurde
10. Jetzt kopieren wir die erstellte ISO "C:\WIN7PE\ISO\Win7PE\_x64.ISO" in unseren Gemeinsamen Ordner "PXE/images/win7pe"
11. Als Letztes fügen wir noch den folgenden Menüeintrag an das Ende unserer default PXE-Konfigurationsdatei an:



```

LABEL win7pe
  MENU LABEL Win7PE x64
  LINUX memdisk
  APPEND iso raw
  INITRD images/win7pe/Win7PE_x64.ISO

```

- **Vorsicht:** Wer anstatt der x64 Version eine x86 Version verwendet hat, muss dies auch im Menüeintrag abändern!

## Windows 7 / 8 / 8.1 / 10

Mit dieser Methode lässt sich ein beliebiges Windows 7, 8, 8.1 oder 10 ISO bzw. DVD von der DiskStation aus booten und installieren. Alles was wir hierzu benötigen ist eine ISO oder DVD von einem Windows 7, 8, 8.1 oder 10 und die passende WinPE.iso, die ich für euch schon erstellt und bereitgestellt habe. Wer es vorzieht seine eigene WinPE.iso zu erstellen, da er evtl. Treiber oder Programme einbinden will/muss, der folgt den folgenden Links zu den passenden HowTo's:

- HowTo WinPE.iso (WinPE 3.0) für Windows 7 erstellen ([http://www.gtkdb.de/index\\_17\\_548.html](http://www.gtkdb.de/index_17_548.html))
- HowTo WinPE.iso (WinPE 3.1) für Windows 7 mit SP1 erstellen ([http://www.gtkdb.de/index\\_17\\_1197.html](http://www.gtkdb.de/index_17_1197.html))
- HowTo WinPE.iso (WinPE 4.0) für Windows 8 erstellen ([http://www.gtkdb.de/index\\_27\\_1812.html](http://www.gtkdb.de/index_27_1812.html))
- HowTo WinPE.iso (WinPE 4.1 bzw. 5.0) für Windows 8.1 erstellen ([http://www.gtkdb.de/index\\_37\\_2367.html](http://www.gtkdb.de/index_37_2367.html))

1. Erstellen eines neuen Ordners "winpe" in "PXE/images/"
2. Als nächstes suchen wir uns einen Ort, wo wir unsere Windows ISOs bzw. DVDs speichern wollen. Ihr könnt dafür einen neuen Gemeinsamen Ordner erstellen oder den vorhandenen "PXE" verwenden und sie z.B. in "PXE/images/isos/" speichern. Ich entpacke als Beispiel meine "Windows 7 Home Premium x64.iso" nach "PXE/images/isos/windows7x64/". **Wichtig:** Diesen Pfad merken, diesen müsst ihr später per Hand in einer Windows-Eingabeaufforderung als Netzlaufwerk einbinden, damit Ihr euer Setup ausführen und starten könnt!
3. Jetzt benötigen wir, wie schon erwähnt, das passende WinPE.iso zum jeweiligen Windows. Wie gesagt, entweder Ihr erstellt eure eigenen oder nehmt meine, die allerdings ohne zusätzliche Treiber oder Programme erstellt wurden:
  - WinPE\_7x86.iso (ca. 136 MB) (<http://ul.to/gvrwnfyf>) (WinPE 3.0)
  - WinPE\_7x64.iso (ca. 160 MB) (<http://ul.to/509v225v>) (WinPE 3.0)
  - WinPE\_7x64.iso (ca. 174 MB) (<http://ul.to/uj1ew1dg>) (WinPE 3.0 erweitert um "Dell WinPE 3.0 Driver CAB Pack" für Netzwerkkarten)
  - WinPE\_8x86.iso (ca. 169 MB) (<http://ul.to/tj0229r0>) (WinPE 4.0)
  - WinPE\_8x64.iso (ca. 201 MB) (<http://ul.to/fsi7n7mi>) (WinPE 4.0)
  - WinPE\_8.1x86.iso (ca. 168 MB) (<http://ul.to/6f1z29w7>) (WinPE 4.1)
  - WinPE\_8.1x64.iso (ca. 203 MB) (<http://ul.to/nmp7g51v>) (WinPE 4.1)
  - WinPE\_10x86.iso (ca. 197 MB) (<http://ul.to/ek6h08vv>) (WinPE 5.0)
  - WinPE\_10x64.iso (ca. 251 MB) (<http://ul.to/xjhrn5nk>) (WinPE 5.0)

1. Diese kopieren wir jetzt in unseren erstellten Ordner "PXE/images/winpe"
2. Als Nächstes fügen wir den jeweiligen Menüeintrag an das Ende unserer default PXE-Konfigurationsdatei an:

```

LABEL winpe_7x86
  MENU LABEL WinPE 7x86
  LINUX memdisk
  APPEND iso raw
  INITRD images/winpe/WinPE_7x86.iso

LABEL winpe_7x64
  MENU LABEL WinPE 7x64
  LINUX memdisk
  APPEND iso raw
  INITRD images/winpe/WinPE_7x64.iso

LABEL winpe_8x86
  MENU LABEL WinPE 8x86

```

Wir Booten die Architektur von WinPE über unser PXE-Bootmenü, die auch unser zu installierendes Windows 7, 8, 8.1 oder 10 hat. Bei mir also WinPE 7x64. Jetzt erscheint nur eine Windows-Eingabeaufforderung. Hier geben wir jetzt folgendes ein:

```
net use v: \\192.168.178.250\PXE\images\isos\windows7x64
```

Das bindet ein Netzlaufwerk als neues Laufwerk "v:\" ein. Ihr achtet natürlich auf die korrekte IP-Adresse eurer DiskStation und auf den von euch gewählten Pfad zum richtigen Windows Image.

Jetzt geben wir noch folgendes ein:

```
v:\setup.exe
```

Jetzt sollte sich das Windows Setup öffnen und wir können wie gewohnt Windows über den Assistenten installieren.

### Systemfehler, die beim Einbinden des Netzlaufwerks entstehen können:

- Systemfehler 55: Ihr habt euch beim angegebenen Pfad vertan. Achtet auf Groß- und Kleinschreibung, die richtigen Backslashes "\" und auf die richtige IP-Adresse eurer DiskStation
- Systemfehler 58: Er konnte das Netzlaufwerk verbunden werden, da kein Nutzer angegeben wurde (Rechteproblem!) -> /user:"benutzer" "passwort" verwenden!
- Systemfehler 67: Ihr hab wohl den Gemeinsamen Ordner, in denen die Windows Images liegen, in der DSM-Systemsteuerung für die Netzwerkumgebung verborgen. Das darf nicht der Fall sein.

### WinPE\_8.1x64 und WinPE\_10x64 Fehler:

Sollte bei euch ein Fehler beim Laden des WinPE\_8.1x64 oder WinPE\_10x64 Images kommen der ungefähr so aussieht:

```

Your PC needs to restart.
Please hold down the power button.
Error Code: 0x0000005D
Parameters:
0x000000000FEBFBFF
0x0000000020000800
0x0000000000000000
0x0000000000000000

```

solltet Ihr prüfen ob Ihr die Windows 8.1 bzw. 10 Voraussetzungen erfüllt:

- Ihr habt mindestens 4 GB RAM
- Ihr habt mindestens 2 Cores (CPU)
- Eure CPU unterstützt: Physical Address Extension (PAE), NX-Bit (No eXecute) bzw. XD-Bit (eXecute Disable) und SSE2 (Teilweise im BIOS explizit einstellbar!)
  - Für die VirtualBox: Unter System/Prozessor muss PAE/NX aktiviert sein und unter System/Beschleunigung muss VT-x/AMD-V und Nested Paging aktiviert sein!
  - Außerdem muss die VirtualBox bei Version auf Windows -> Windows 8.1 bzw. 10 (64bit) eingestellt sein!

### Erstellen einer eigenen WinPE.iso mit verändertem Startskript

Hier beschreibe ich wie wir eine eigene WinPE.iso erstellen und das Startskript verändern, um zb. den Befehl "net use" und das Aufrufen der setup.exe einzubinden. Da wir das WAIK von Microsoft installieren müssen empfehle ich hierfür eine virtuelle Maschine zu verwenden, um unser System nicht unnötig vollzumüllen. Ich beschreibe dies hierfür nur am Beispiel für Windows 7 x64 SP1 anhand der oben geposteten Anleitung zum Erstellen einer WinPE 3.1 ISO. Für die anderen Architekturen und Versionen (Windows 8) setze ich eine gewisse Transferleistung voraus (Verwenden der anderen Anleitungen), weise aber an den jeweiligen Stellen noch mal explizit darauf hin. Das verwendete Betriebssystem wo wir unser WAIK installieren und unsere WinPE.iso erstellen sollte egal sein! Um sicher zu gehen verwende ich aber das gleiche, wie das wofür unsere WinPE.iso gedacht ist, sprich Windows 7 x64 SP1.

#### 1. Installieren des WAIK:

1. Herunterladen und installieren vom WAIK für Windows 7 (Hinweis: Für andere WinPE.iso's für andere Architekturen bzw. Betriebssysteme sind andere WAIK's erforderlich. Siehe die jeweiligen Anleitungen oben!)
  - WAIK für Windows 7 (ca. 1,6 GB) (<http://www.microsoft.com/de-de/download/details.aspx?id=5753>) Verwendeter Programmpfad: "C:\Program Files\Windows AIK\"
2. Herunterladen der Erweiterung vom WAIK für Windows 7 und einlegen bzw. mounten der ISO (Bei mir Laufwerk "E:", später bei den Pfadangaben in den Befehlen berücksichtigen!)
  - WAIK Erweiterung für Windows 7 SP1 (ca. 1,3 GB) (<http://www.microsoft.com/de-de/download/details.aspx?id=5188>)
3. Öffnen der Eingabeaufforderung (cmd) als Administrator (Start -> "cmd" eingeben, Rechtsklick -> Als Administrator ausführen)
4. Zum aktualisieren des installierten WAIK für SP1 kopieren wir jetzt die eingelegte DVD in den Installationspfad mit folgendem Befehl:

```
xcopy E:\*.* /s /e /f "C:\Program Files\Windows AIK\Tools\PETools"
```

- Wir bestätigen bei der Aufforderung zum Überschreiben der vorhandenen Dateien mit "A" für Alle

#### 2. Vorbereiten der WinPE.iso

- Mit Hilfe des cypype.cmd Tools erstellen wir einen neuen Ordner und kopieren die für WinPE nötigen Dateien hinein. Dazu geben wir folgenden Befehl in unsere, mit Administrator-Rechten bereits geöffnete, Eingabeaufforderung ein:

```
"C:\Program Files\Windows AIK\Tools\PETools\copyype.cmd" amd64 C:\WinPE_x64
```

Hinweis: schaut in "C:\Program Files\Windows AIK\Tools\PETools\" welche Architekturen ihr verwenden könnt! Danach richtet sich euer Befehl. Verwendet die Architektur, für die euer WinPE.iso später gedacht ist!

### 3. Bearbeiten der winpe.wim und des darin enthaltenen Startskripts

1. Prüfen ob der Ordner "mount" unter "C:\WinPE\_x64\" bereits vorhanden und leer ist. Wenn nicht erstellen!
2. Wir öffnen nun, wieder als Administrator, die "**Eingabeaufforderung für Bereitstellungstools**" unter "Start / Alle Programme / Microsoft Windows AIK". **Wichtig:** die normale Eingabeaufforderung, die wir auch vorher verwendet haben, funktioniert hier nicht richtig, da die jetzt verwendeten Befehle wie "dism" hier abgeändert wurden bzw. in einer anderen Version vorliegen!
3. Mounten der winpe.wim in den Ordner "C:\WinPE\_x64\mount\" mit folgendem Befehl:

```
dism /mount-wim /wimfile:C:\WinPE_x64\winpe.wim /index:1 /MountDir:C:\WinPE_x64\mount
```

4. Wir haben nun das winpe.wim File gemountet und können unser Startskript nun bearbeiten. Dazu öffnen wir den Editor als Administrator und öffnen im Editor unser Skript (Rechtsklick auf das Skript selber und dann "Bearbeiten" wird nicht funktionieren, da wir es so nicht als Administrator öffnen und anschließend auch nicht speichern können!) unter:

```
"C:\WinPE_x64\mount\Windows\System32\startnet.cmd"
```

- **Wichtig:** Der darin schon enthaltene Befehl "wpeinit" muss dabei zwingend als erster Befehl drin stehen bleiben
- Wir können hier nun alles eintragen was wir möchten (net use, ping, etc ...). Da das spätere Verändern des Startskriptes wie ihr sicher merkt einen mega Aufwand bedeutet, bzw. ein erneutes Erstellen einer WinPE.iso erfordert, empfehle ich hier keine Eintragungen außer den "net use"-Befehl zu machen. Dann lassen wir das Skript einfach ein anderes aufrufen, dass auf unserem Netzlaufwerk bzw. NAS liegt. So können wir später das Skript auf unserer NAS beliebig oft und einfach verändern, ohne das WinPE nochmals anfassen zu müssen! In unsere startnet.cmd schreiben wir also folgendes:

```
wpeinit
```

```
net use v: \\192.168.178.250\PXE\images\isos /user:pxe pxe
```

```
v:\windows7x64.cmd
```

Die Angabe eines Benutzers mit Passwort ("/user:pxe pxe") ist nicht zwingend erforderlich. Sollte nur aufzeigen wie es geht, wenn man es braucht.

- Skript abspeichern, Editor schließen und auch aus dem Ordner "C:\WinPE\_x64\" rausgehen, damit ein reibungsloses unmounten möglich ist und keine Fehler bringt!
5. Unmounten der winpe.wim mit folgendem Befehl in der Eingabeaufforderung für Bereitstellungstools (nicht durcheinander kommen mit seinen ganzen Editoren und Eingabeaufforderungen!):

```
dism /unmount-wim /mountdir:C:\WinPE_x64\mount /commit
```

6. Die Eingabeaufforderung für Bereitstellungstools kann nun geschlossen werden. **Wir fahren wieder fort mit unserer normalen Eingabeaufforderung!**

### 4. Erstellen der WinPE.iso

1. Kopieren und umbenennen der winpe.wim in den ISO Ordner mit:

```
copy C:\WinPE_x64\winpe.wim C:\WinPE_x64\ISO\sources\boot.wim
```

2. Erstellen der ISO mit Hilfe von oscdimg.exe:

```
"C:\Program Files\Windows AIK\Tools\amd64\oscdimg.exe" -n -bC:\WinPE_x64\etfsboot.com C:\WinPE_x64\ISO C:\WinPE_x64\WinPE_7x64_
```

5. Unsere WinPE\_7x64\_SP1.iso befindet sich nun im Ordner C:\WinPE\_x64 und kann in unseren Gemeinsamen Ordner "PXE\images\winpe" kopiert werden
6. Nun erstellen wir unser Skript, das nun die setup.exe ausführt in unserem Gemeinsamen Ordner "PXE\images\isos" und nennen es "windows7x64.cmd" mit folgendem Inhalt:

```
v:\windows7x64\setup.exe
```

Wer möchte kann hier nun eintragen was er will, was automatisch beim booten des WinPE.iso's über PXE ausgeführt werden soll!

7. Fertig!

**Hinweis:** Schritt 3 ist der einzige, der von den oben geposteten Anleitungen abweicht und kann, unter Voraussetzung der genannten Transferleistung, so auf alle Betriebssysteme bzw. WinPE.iso's angewendet werden!

## Erstellen einer eigenen WinPE.iso mit verändertem Startskript (Windows 10 only)

Beim Erstellen einer eigenen WinPE.iso für Windows 10 gibt es 2-3 kleinere Abweichungen von meiner obigen Anleitung auf die ich hier kurz eingehen will.

### 1. Installieren des Windows 10 ADK:

#### 1. Herunterladen vom Windows 10 ADK Installer

- ADK Installer für Windows 10 (1,8 MB der Installer, ca. 3,3 GB Download) (<http://go.microsoft.com/fwlink/?LinkId=526740>)

#### 2. Beim Installieren des Installers wählen wir folgende Inhalte aus:

- Wir installieren standardmäßig nach: "C:\Program Files (x86)\Windows Kits\10\" (Wer das nicht macht, muss später das Script manuell abändern!!!)
- **Bereitstellungstools** (64,2 MB)
- **Windows-Vorinstallationsumgebung (Windows PE)** (3,2 GB)
- der Rest an Tools kann abgewählt werden

#### 3. Das jetzt von Microsoft installierte script "cotype.cmd", das wir zum Kopieren der WinPE Dateien benötigen ist fehlerhaft und beinhaltet falsche Pfadangaben. (Stand: August 2015) Ich hab es für euch abgeändert und bereitgestellt. Nochmals der Hinweis: Wer den Standard-Installationspfad nicht verwendet hat, muss das Script nochmal durchschauen und die Pfade entsprechend abändern!

- windows10\_adk\_cotype.cmd.zip (1,3 KB) (<http://ul.to/xfsovs10>)
- Entpacken und mit der vorhandenen Datei ersetzen in

```
"C:\Program Files (x86)\Windows Kits\10\Assessment and Deployment Kit\Windows Preinstallation Environment\"
```

### 2. Vorbereiten der WinPE.iso (Hier als Beispiel mit x64 Architektur)

- Mit Hilfe des jetzt richtigen cotype.cmd Tools kopieren wir die für WinPE nötigen Dateien nach "C:\". Dazu geben wir folgenden Befehl in eine, mit **Administrator-Rechten geöffnete, Eingabeaufforderung** ein:

```
"C:\Program Files (x86)\Windows Kits\10\Assessment and Deployment Kit\Windows Preinstallation Environment\cotype.cmd" amd64 C:\
```

### 3. Bearbeiten der boot.wim und des darin enthaltenen Startskripts

1. Wir öffnen nun, wieder als Administrator, die **"Eingabeaufforderung für Bereitstellungstools"**
2. Mounten der boot.wim in den Ordner "C:\WinPE\_x64\mount\" mit folgendem Befehl:

```
dism /mount-wim /wimfile:C:\WinPE_x64\media\sources\boot.wim /index:1 /MountDir:C:\WinPE_x64\mount
```

#### 3. Nun verändern wir das Startscript wie gehabt unter:

```
"C:\WinPE_x64\mount\Windows\System32\startnet.cmd"
```

- Mit einer, als Administrator, geöffneten Konsole jetzt eintragen was man möchte. "wpeinit" als ersten Eintrag stehen lassen und nach meiner Erfahrung ist die Angabe von /user beim "net use"-Befehl mittlerweile Pflicht, da ich sonst einen Fehler 58 bekomme! Also legt euch einen neuen Nutzer an so wie ich, der Lese- und Ausführrechte auf den Ordner isos hat!

```
wpeinit
```

```
net use v: \\192.168.178.250\pxe\images\isos /user:pxe pxe
```

```
v:\windows10x64.cmd
```

#### 4. Unmounten der boot.wim mit folgendem Befehl in der Eingabeaufforderung für Bereitstellungstools:

```
dism /unmount-wim /mountdir:C:\WinPE_x64\mount /commit
```

### 4. Erstellen der WinPE.iso

#### 1. Erstellen der ISO mit Hilfe von oscdimg.exe:

```
"C:\Program Files (x86)\Windows Kits\10\Assessment and Deployment Kit\Deployment Tools\amd64\Oscdimg\oscdimg.exe" -n -bC:\WinPE\
```

2. Unsere WinPE\_10x64.iso befindet sich nun im Ordner C:\WinPE\_x64 und kann in unseren Gemeinsamen Ordner "PXE\images\winpe" kopiert werden
5. Nun erstellen wir unser Skript, das nun die setup.exe ausführt in unserem Gemeinsamen Ordner "PXE\images\isos" und nennen es "windows10x64.cmd" mit folgendem Inhalt:

```
v:\windows10x64\setup.exe
```

6. Fertig!

3x3cut0r 15:42, 19. Jun. 2014 (CEST)

Von „<http://www.synology-wiki.de/index.php?title=PXE&oldid=6931>“

- 
- Diese Seite wurde zuletzt am 23. September 2015 um 20:21 Uhr geändert.