

Kernel anpassen

- Arbeitsverzeichnis erstellen und reinwechseln
 - `mkdir uebung && cd uebung`
- auf kernel.org gehen und URL des aktuellsten Kernels notieren
- Kernel runterladen. Z.B.
 - `wget https://cdn.kernel.org/pub/linux/kernel/v5.x/linux-5.0.7.tar.xz`
- Kernel auspacken
 - `tar xJvf linux-5.0.7.tar.xz`
- ins Kernel Quellcode-Verzeichnis wechseln
 - `cd linux-5.0.7`
- Abhängigkeiten des Kernel Builds installieren:
 - unter Debian/Ubuntu/etc.
 - `apt-get install ncurses-dev pkg-config bc gcc libc6-dev`**
 - `make bzip2 binutils dpkg-dev flex bison`
 - unter Fedora/RedHat/CentOS/etc,
 - `yum install bison flex bc rpm-build bc gcc make bzip2 ncurses-devel # nicht getestet`
- Kernel konfigurieren
 - eigenen Kernel konfigurieren
 - `make menuconfig`
 - so viel Unnötiges wie mögliche weg-konfigurieren, damit Bauzeit kürzer wird
 - VirtualBox Netzwerk-Karte ist Intel PRO/1000
 - minimalen Kernel konfigurieren
 - `make tinyconfig`
- `vim +421 arch/x86/boot/compressed/misc.c`
 - folgende Zeile finden:
 - `debug_putstr("done.nBooting the kernel.n");`
 - Meldung frei anpassen
- Paket des Kernels bauen
 - Debian
 - als root: `make deb-pkg`
 - dauert lange
 - rpm
 - als root: `make binrpm-pkg`
 - dauert lange
 - Paket ist unter `/root/rpmbuild/RPMS/$ARCH/kernel-5.0.7-1.i386.rpm`

- wenn nötig neuen Kernel in VM hineinkopieren
 - Parameter des folgenden Kommandos müssen an lokale Gegebenheiten angepasst werden:
 - `scp -P 1234567 ../linux-image-5.0.7_5.0.7-1_amd64.deb localhost:/tmp`
- neuen Kernel in VM installieren
 - `cd /dorthin_wo_linux-image-5.0.7_5.0.7-1_amd64.deb_ist` (entweder unter /tmp oder ../)
 - Debian
 - `dpkg -i linux-image-5.0.7_5.0.7-1_amd64.deb`
 - rpm
 - `rpm -i kernel-5.0.7-1.i386.rpm`
 - `vim /etc/grub.d/40_custom`
 - menu entry hinzufügen, analog zu /boot/grub2/grub.conf
 - `grub2-mkconfig -o /boot/grub2/grub.cfg`
- VM neustarten