LPIC-2

Michael Dienert

30. April 2013

Inhaltsverzeichnis

201 Linux Kernel **3**
- 201.1 Kernel Components . . . 3
- 201.2 Compiling a kernel . . . 4
- 201.3 Patching a kernel . . . 4
- 201.4 Customise, build and install a custom kernel and kernel modules . . . 5
- 201.5 Manage/Query kernel and kernel modules at runtime . . . 5

202 System Startup **8**
- 202.1 Customising system startup and boot processes . . . 8
- 202.2 System recovery . . . 11

203 Filesystem and Devices **12**
- 203.1 Operating the Linux filesystem . . . 12
- 203.2 Maintaining a Linux filesystem . . . 14
- 203.3 Creating and configuring filesystem options . . . 16
- 203.4 udev Device Management . . . 17

204 Advanced Storage Device Administration **19**
- 204.1 Configuring RAID . . . 19
- 204.2 Adjusting Storage Device Access . . . 20
- 204.3 Logical Volume Manager . . . 21

205 Networking Configuration **23**
- 205.1 Basic networking configuration . . . 23
- 205.2 Advanced Network Configuration and Troubleshooting . . . 24
- 205.3 Troubleshooting network issues . . . 25
- 205.4 Notify users on system-related issues . . . 28

206 System Maintenance **28**
- 206.1 Make and install programs from source . . . 28
- 206.2 Backup operations . . . 29

207 Domain Name Server **30**
207.1 Basic DNS server configuration 30
207.2 Create and maintain DNS zones 31
207.3 Securing a DNS server 34

208 Web Services **35**
208.1 Implementing a web server 35
208.1.1 Apache-Direktiven 35
208.1.2 Globale Konfiguration 37
208.1.3 Nur bestimmte Client-Adressen zulassen 40
208.1.4 Alias und Weiterleitung 40
208.1.5 .htaccess: hypertext access 41
208.1.6 Dynamische Webseiten 44
208.2 Maintaining a web server 46
208.2.1 Virtual Hosts 46
208.2.2 SSL 48
208.3 Implementing a proxy server 49
208.3.1 ACL 50
208.3.2 Filterregeln 50

209 File Sharing **52**
209.1 SAMBA Server Configuration 52
209.1.1 smbd, nmbd 52
209.1.2 Samba als PDC 54
209.1.3 testparm 54
209.1.4 smbpasswd 55
209.1.5 smbstatus 56
209.1.6 nmblookup 56
209.1.7 smbclient 56
209.1.8 smbmount 56
209.1.9 Das net-Kommando 57
209.2 NFS Server Configuration 57
209.2.1 /etc/exports 57
209.2.2 exportfs und nfs-kernel-server 58
209.2.3 Portmapper und mountd 58
209.2.4 Weitere wichtige Kommandos 59
209.2.5 NFS-Client 59

210 Network Client Management **59**
210.1 210.1 DHCP configuration 59
210.1.1 ISC-dhcp-server 60
210.1.2 dhclient 60
210.1.3 dhcpd.leases 60

210.2210.2 PAM authentication 61
210.2.1 PAM 61
210.3210.3 LDAP client usage 64

211211: E-Mail Services **64**
211.1211.1 Using e-mail servers 64
211.1.1 postfix-Konfiguration 65
211.2211.2 Managing Local E-Mail Delivery 65
211.2.1 procmail 66
211.3211.3 Managing Remote E-Mail Delivery 67
211.3.1 dovecot.conf 68

212Topic 212: System Security **68**
212.1212.1 Configuring a router 68
212.1.1 iptables 69
212.1.2 iptables sichern und restaurieren 73
212.2212.2 Securing FTP servers 73
212.2.1 /etc/vsftpd.conf 73
212.3212.3 Secure shell (SSH) 73
212.3.1 sshd_config 75
212.3.2 Schlüsselverwaltung 76
212.4212.4 TCP Wrapper 77
212.5212.5 Security tasks 77

213Topic 213: Troubleshooting **77**
213.1213.1 Identifying boot stages and troubleshooting bootloaders 77
213.2213.2 General troubleshooting 78
213.3213.3 Troubleshooting system resources 78
213.4213.4 Troubleshooting environment configurations 79

201 Linux Kernel

201.1 Kernel Components

Weight: 2

Description: Candidates should be able to utilise kernel components that are necessary to specific hardware, hardware drivers, system resources and requirements. This objective includes implementing different types of kernel images, identifying stable and development kernels and patches, as well as using kernel modules.

Key Knowledge Areas

Kernel 2.6.x documentation

Terms and Utilities

```
    /usr/src/linux
    /usr/src/linux/Documentation
    zImage  (gzip-komprimiert, alt)
    bzImage (big-zImage, neu)
```

```
file /boot/vmlinuz-3.2.0-29-generic-pae
   /boot/vmlinuz-3.2.0-29-generic-pae:
   Linux kernel x86 boot executable bzImage,
   version 3.2.0-29-generic-pae (buildd@roseapple) #46-Ubuntu SMP Fri Jul ,
   RO-rootFS, swap_dev 0x4, Normal VGA

uname -v
   #46-Ubuntu SMP Fri Jul 27 17:25:43 UTC 2012
uname -r
   3.2.0-29-generic-pae
```

201.2 Compiling a kernel

Weight: 2

Description: Candidates should be able to properly configure a kernel to include or disable specific features of the Linux kernel as necessary. This objective includes compiling and recompiling the Linux kernel as needed, updating and noting changes in a new kernel, creating an initrd image and installing new kernels.

Key Knowledge Areas

`/usr/src/linux/` GRUB configuration files Kernel 2.6.x make targets

Terms and Utilities

```
    mkinitrd
    mkinitramfs -o ~/tmp/initramfs-$(uname -r)
    mkinitramfs -k -o ~/tmp/initramfs-2.6.21-686 2.6.21-686  // parameter ist kernel-version
    make
    make targets
    config
    xconfig
    menuconfig
    oldconfig
    mrproper
    zImage
    bzImage
    modules
    modules_install
```

mkinitramfs erzeugt `initramfs`. `initramfs` ist ein ein komprimiertes cpio-Archiv und ist der Nachfolger von `mkinitrd`. Beide verwenden die gleiche Syntax. Wird keine Kernel-Version angegeben, wird der aktuelle Kernel verwendet

201.3 Patching a kernel

Weight: 1

Description: Candidates should be able to properly patch a kernel to add support for new hardware. This objective also includes being able to properly remove kernel patches from already patched kernels.

Key Knowledge Areas

Kernel Makefiles

Terms and Utilities

```
patch
gzip
bzip2
```

Beispiel: Nach Aufruf vom `patch`-Kommando sind die Dateien `original.txt` und `neueVersion.txt` **identisch**:

```
diff original.txt neueVersion.txt > patchDatei.txt
patch original.txt patchDatei.txt
```

Rückgängig machen des Patches:

```
patch -R original.txt patchDatei.txt
```

201.4 Customise, build and install a custom kernel and kernel modules

Weight: 2

Description: Candidates should be able to customise, build and install a 2.6 kernel for specific system requirements, by patching, compiling and editing configuration files as required. This objective includes being able to assess requirements for a kernel compile as well as build and configure kernel modules.

Key Knowledge Areas

Customize the current kernel configuration. Build a new kernel and appropriate kernel modules. Install a new kernel and any modules. Ensure that the boot manager can locate the new kernel and associated files.

`/usr/src/linux/`

Module configuration files

Terms and Utilities

```
patch
make
module tools
/usr/src/linux/*
/usr/src/linux/.config
/lib/modules/kernel-version/*
/boot/*
```

201.5 Manage/Query kernel and kernel modules at runtime

Weight: 2

Description: Candidates should be able to manage and/or query a 2.6.x kernel and its loadable modules.

Key Knowledge Areas

Use command-line utilities to get information about the currently running kernel and kernel modules. Manually load and unload kernel modules. Determine when modules can be unloaded. Determine what parameters a module accepts. Configure the system to load modules by names other than their file name.

Terms and Utilities

```
    /lib/modules/kernel-version/modules.dep
    module configuration files in /etc
    /proc/sys/kernel/
    depmod
    insmod
    lsmod
    rmmod
    modinfo
    modprobe
    uname
```

depmod Erzeugt die Datei `/lib/modules/`uname -r`/modules.dep.bin`

insmod, lsmod, rmmod einfachste Tools zum Einfügen, Auflisten oder Entfernen von Modulen. `lsmod` zeigt auch Abhängigkeiten auf.

/proc/modules Datei, die die gleiche Info wie `lsmod` enthält

modprobe

modprobe

Usage:

```
  modprobe [-v]  [-V]  [-C config-file]  [-n]  [-i]  [-q]  [-b]
           [modulename]  [module parameters ...]
  modprobe [-l]  [-t dirname]  [wildcard]
  modprobe [-r]  [-v]  [-n]  [-i]  [modulename ...]   //Modul entfernen
  modprobe [-c]                       //Aktuelle Konfiguration ausgeben

-v verbose
-V version
-n dry-run
-l module in /lib/modules/`uname -r` auflisten
-i install/remove-Kommandos in der conig-datei ignorieren
-b blacklist-Kommandos auch auf die Modulnamen anwenden

Beispiele:

modprobe -t net  //versuche solange Netzwerkmodule zu laden,
                   bis eins funktioniert
modprobe -at net //alle Netzwerkmodule laden
modprobe -lt net //alle Netzwerkmodule auflisten.
                   Achtung: -tl funzt nicht !!!
```

`modprobe` fügt Module *intelligent* zum Kernel hinzu oder entfernt diese. Es wertet die Dateien

```
/lib/modules/`uname -r`/modules.dep.bin
/etc/modules.conf
/etc/modprobe.d/*
```

aus.

`modules.dep.bin` wird mit `depmod` erzeugt. Module werden in `/lib/modules/`uname -r`/` gesucht.

Beispiel für modules.conf:

```
alias parport_lowlevel    parport_pc
options parport_pc io=0x378 irq=none,none

options ne      io=0x300

alias block-major-1       rd
alias block-major-2       floppy

ptions bttv    pll=1 radio=0 card=0
post-install bttv /sbin/modprobe "-k" tuner;

options dummy0 -o dummy0
options dummy1 -o dummy1

# ppp over ethernet
# the kernel 2.2 uses pppox
# the kernel 2.4 uses pppoe
if `kernelversion` == "2.2"
alias char-major-144      pppox
post-install pppox insmod mssclampfw
pre-remove   pppox rmmod  mssclampfw
else
alias char-major-108      ppp_async
alias char-major-144      pppoe
alias net-pf-24           pppoe
endif

# agpgart is i386 only right now
pre-install mga /sbin/modprobe "-k" "agpgart"
pre-install r128 /sbin/modprobe "-k" "agpgart"
pre-install radeon /sbin/modprobe "-k" "agpgart"
options agpgart agp_try_unsupported=1
```

Bedeutung der Einträge:

alias `alias wildcard modulename` Beispiel: `alias eth0 r8169`

options `options modulename opt1=val1 opt2=val2...` Der Modulname kann auch ein *alias* sein. Modulname ohne Endung .ko (.so)

install, remove, pre-install/remove, post-install/remove `install modulename command` .

Für das entsprechende Modul wird bei der Installation das angegebene Shellkommando ausgeführt und nicht das für die Installation vorgesehene Kommando.

Entsprechendes gilt für pre-/post- -install/remove.

path `path=Pfad` Pfad, in dem nach Modulen gesucht wird.

blacklist `blacklist modulename` Module könenn interne Aliasnamen haben. Der blacklist-Eintrag zeigt an, dass alle internen Aliasnamen des entsprechenden Moduls *ignoriert* werden sollen.

202 System Startup

202.1 Customising system startup and boot processes

Weight: 4

Description: Candidates should be able to query and modify the behaviour of system services at various run levels. A thorough understanding of the init structure and boot process is required. This objective includes interacting with runlevels.

Key Knowledge Areas

Linux Standard Base Specification (LSB)

Terms and Utilities

```
/etc/inittab
/etc/init.d/
/etc/rc.d/
chkconfig
update-rc.d
```

inittab

`/etc/inittab` enthält eine Tabelle mit 4 Spalten.

```
# /etc/inittab

# default runlevel
id:2:initdefault:

# check system on startup
# first script to be executed if not booting in emergency (-b) mode
si:I:bootwait:/sbin/init.d/boot

# /sbin/init.d/rc takes care of runlevel handling
#
# runlevel 0 is halt
# runlevel S is single-user
# runlevel 1 is multi-user without network
# runlevel 2 is multi-user with network
# runlevel 3 is multi-user with network and xdm
# runlevel 6 is reboot
l0:0:wait:/sbin/init.d/rc 0
l1:1:wait:/sbin/init.d/rc 1
l2:2:wait:/sbin/init.d/rc 2
l3:3:wait:/sbin/init.d/rc 3
l6:6:wait:/sbin/init.d/rc 6

# what to do in single-user mode
ls:S:wait:/sbin/init.d/rc S
~~:S:respawn:/sbin/sulogin

# what to do when CTRL-ALT-DEL is pressed
ca::ctrlaltdel:/sbin/shutdown -r -t 4 now

# what to do when power fails/returns
pf::powerwait:/sbin/init.d/powerfail    start
pn::powerfailnow:/sbin/init.d/powerfail now
po::powerokwait:/sbin/init.d/powerfail  stop

# getty-programs for the normal runlevels
# <id>:<runlevels>:<action>:<process>
# The "id" field  MUST be the same as the last
# characters of the device (after "tty").
1:123:respawn:/sbin/mingetty --noclear tty1
2:123:respawn:/sbin/mingetty tty2
3:123:respawn:/sbin/mingetty tty3
4:123:respawn:/sbin/mingetty tty4
5:123:respawn:/sbin/mingetty tty5
6:123:respawn:/sbin/mingetty tty6

# end of /etc/inittab
```

Spalte 1: eindeutige ID. Ein oder zwei Zeichen.

Spalte 2: Nummer des oder der Runlevel. Wenn das Feld leer ist, gilt die Zeile für alle Runlevel.

Spalte 3: Aktion:

boot Startet, wenn die /etc/inittab zum ersten Mal gelesen wird

bootwait Startet nach den Einträgen für boot

ctrlaltdel Startet nach der Tasteneingabe <Strg>+<Alt>+<Entf>

initdefault Definiert den Standard-Runlevel

once Startet beim Wechsel eines Runlevels

ondemand Hält Prozesse in bestimmten Runlevels am Laufen

powerfail Startet, wenn ein Stromausfallsignal empfangen wurde

sysinit Startet, bevor die Einträge boot und bootwait ausgeführt werden.

respawn Startet den Prozeß nach seinem Ende neu

wait Startet den Prozeß einmal beim Erreichen des Runlevels und init wartet auf sein Ende

Spalte 4: Kommando mit Pfad und Optionen

chkconfig

```
chkconfig -l apache postgresql acpi-support
apache2                   0:off  1:off  2:on   3:on   4:on   5:on   6:off
postgresql                0:off  1:off  2:on   3:on   4:on   5:on   6:off
acpi-support              0:off  1:off  2:on   3:on   4:on   5:on   6:off

chkconfig apache 35 //configure the apache web server for runlevel 3 and 5.
chkconfig foo 2 //
```

update-rc.d

```
update-rc.d [-n] [-f] <basename> remove
       update-rc.d [-n] <basename> defaults [NN | SS KK]
       update-rc.d [-n] <basename> start|stop NN runlvl [runlvl] [...] .
       update-rc.d [-n] <basename> disable|enable [S|2|3|4|5]
                -n: not really
                -f: force

update-rc.d foo defaults 99  //achtung: /etc/init.d/foo muss existieren
Adding system startup for /etc/init.d/foo ...
   /etc/rc0.d/K99foo -> ../init.d/foo
   /etc/rc1.d/K99foo -> ../init.d/foo
   /etc/rc6.d/K99foo -> ../init.d/foo
   /etc/rc2.d/S99foo -> ../init.d/foo
   /etc/rc3.d/S99foo -> ../init.d/foo
   /etc/rc4.d/S99foo -> ../init.d/foo
   /etc/rc5.d/S99foo -> ../init.d/foo

update-rc.d foo stop 99 0 1 6 . start 99 2 3 4 5 .
Adding system startup for /etc/init.d/foo ...
   /etc/rc0.d/K99foo -> ../init.d/foo
   /etc/rc1.d/K99foo -> ../init.d/foo
   /etc/rc6.d/K99foo -> ../init.d/foo
   /etc/rc2.d/S99foo -> ../init.d/foo
   /etc/rc3.d/S99foo -> ../init.d/foo
   /etc/rc4.d/S99foo -> ../init.d/foo
   /etc/rc5.d/S99foo -> ../init.d/foo

update-rc.d foo remove
update-rc.d: /etc/init.d/foo exists during rc.d purge (use -f to force)

rm /etc/init.d/foo
update-rc.d foo remove
 Removing any system startup links for /etc/init.d/foo ...
   /etc/rc0.d/K99foo
   /etc/rc1.d/K99foo
   /etc/rc2.d/S99foo
   /etc/rc3.d/S99foo
   /etc/rc4.d/S99foo
   /etc/rc5.d/S99foo
   /etc/rc6.d/K99foo
```

202.2 System recovery

Weight: 4

Description: Candidates should be able to properly manipulate a Linux system during both the boot process and during recovery mode. This objective includes using both the init utility and init-related kernel options.

Key Knowledge Areas

inittab GRUB grub shell

Terms and Utilities

```
    init
    mount
    fsck
    telinit
```

Aktuelle Kernel-Parameter anschauen

```
cat /proc/cmdline
BOOT_IMAGE=/boot/vmlinuz-3.2.0-29-generic-pae
root=UUID=5930373b-1fc7-4d05-9dbb-6331cfd207ab
ro quiet splash vt.handoff=7
```

Kernelparameter mit Grub2 übergeben

In Datei /etc/default/grub eintragen:

```
GRUB_CMDLINE_LINUX_DEFAULT="quiet splash vga=795"

update-grub //anschliessend ausfuehren,
            //damit wird /boot/grub/grub.cfg neu erzeugt
```

203 Filesystem and Devices

203.1 Operating the Linux filesystem

Weight: 4

Description: Candidates should be able to properly configure and navigate the standard Linux filesystem. This objective includes configuring and mounting various filesystem types.

Key Knowledge Areas

The concept of the fstab configuration Tools and utilities for handling SWAP partitions and files Use of UUIDs

Terms and Utilities

```
    /etc/fstab
    /etc/mtab
    /proc/mounts
    mount and umount
    sync
    swapon     //siehe unten bei mkswap
    swapoff    //siehe unten bei mkswap
```

/etc/fstab und mount-Kommando

Auf `/etc/fstab` wird von Programmen (z.b. mount) nur lesend zugegriffen. Die Datei enthält eine Tabelle mit 6 Spalten. Die Spalten werden durch *Withespace* getrennt:

Spalte 1: was wird gemountet:

Gerätedatei `/dev/sda1`

NFS-Mount `msv.wara.de:/home`

UUID `UUID=5930373b-1fc7-4d05-9dbb-6331cfd207ab`

LABEL `LABEL=Boot`

Spalte 2: hier steht der Pfad zum Mount-Point. Bei Swap: `none`

Spalte 3: Typ des Dateisystems: `autofs, ext2, ext3, hfs, hpfs, iso9660, jfs, minix, msdos, nfs, ntfs, proc, reiserfs, romfs, smbfs, ufs, vfat, xfs` und viele andere.
Dann gibt's noch: `swap, none, ignore`
Obige Typen werden auch beim Kommando `mount -t <typ> ...` verwendet

Spalte 4: hier stehen mount-options:

`defaults` entspricht `rw, suid, dev, exec, auto, nouser, async`

`rw`, `ro` lese- und schreibbar oder eben nur lesbar mounten

`suid`, `nosuid` wirkung des uid und gid-Bits ein-/ausschalten

`dev`

`exec`, `noexec` Ausführbarkeit von Binaries erlauben/verbieten

`auto`, `noauto` mounten/nicht mounten, wenn `mount -a` ausgeführt wird.

`user`, `nouser` mounten für Normaluser erlauben/verbieten. Unmounten darf nur der, der gemountet hat (vgl. `users`)

`users` mounten und unmounten für alle Benutzer erlaubt. Mounter und Unmounter dürfen verschiedene User sein.

`async`, `sync` I/O-Ops zum Dateisystem erfolgen asynchron=verzögert/synchron=sofort

`remount` erneutes Mounten evtl. mit anderen, neuen Optionen

Spalte 5: Wert 0 = kein dump oder 1 = dump. dump ist ein Kommando, as Backups erzeugt.

Spalte 6: **0 oder nix** : nie fsck ausführen

1 : zuerst fsck ausführen, d.h. /-Dateisystem sollte 1 haben

2 : nach 1 ausführen

Beispiele mount/umount-Kommando:

```
mount -t nfs 141.31.147.114:/home/dienert /home/dienert
umount -at nfs   //alle nfs-mounts auswerfen

blkid /dev/sda1
  /dev/sda1: UUID="5930373b-1fc7-4d05-9dbb-6331cfd207ab" TYPE="ext4"
mount UUID=5930373b-1fc7-4d05-9dbb-6331cfd207ab /mnt

mount -o loop ipcop-1.4.20-install-cd.i386.iso /mnt

losetup /dev/loop1 ipcop-1.4.20-install-cd.i386.iso
mount /dev/loop1 /mnt
```

203.2 Maintaining a Linux filesystem

Weight: 3

Description: Candidates should be able to properly maintain a Linux filesystem using system utilities. This objective includes manipulating standard filesystems.

Key Knowledge Areas

Tools and utilities to manipulate and ext2 and ext3 Tools and utilities to manipulate reiserfs V3 Tools and utilities to manipulate xfs

Terms and Utilities

```
    fsck (fsck.*)
    badblocks
    mkfs (mkfs.*)
    dumpe2fs
    debugfs, debugreiserfs
    tune2fs, reiserfstune
    mkswap
    xfs_info
    xfs_check
    xfs_repair
```

fsck und badblocks

Kommandos: `fsck` `fsck.ext2, fsck.ext3, fsck.ext4` sind nur Links auf /sbin/e2fsck

Optionen und Beispiele

```
 -p        automatische Reparatur (keine Fragen)
 -n        keine Veränderungen am Dateisystem vornehmen
 -y        " Ja " auf alle Fragen annehmen
 -c        suche nach defekten Blöcken
 -f        erzwinge die Überprüfung auch wenn alles i.O. erscheint
 -v                  sei gesprächig
 -b Superbloc        Nutze Superblockkopie
 -B Blockgröße       erzwinge Blockgröße beim Suchen vom Superblock
 -j externes-Journal Angabe des Speicherortes des externen Jounals
 -l bad_blocks_file  zur Liste der defekten Blöcke hinzufügen
 -L bad_blocks_file  Liste der defekten Blöcke definieren

fsck.ext4 -nf /dev/sda3

badblocks -v /dev/sda3
Prüfe von Block 0 bis 486527999
Suche nach defekten Blöcken (Nur-Lesen-Modus):
0.20% erledigt, 0:09 verstrichen. (0/0/0 Fehler)
```

mkfs, dumpe2fs

`mkfs` ist ein Frontend für: `mkfs.ext2, mkfs.ext3, mkfs.ext4` usw. Auswahl mit dem Schalter -t.

`mkfs.ext2, mkfs.ext3, mkfs.ext4` sind Links auf `mke2fs`

`dumpe2fs` gibt ext2,3,4-Infos aus. Der Schalter -h begrenzt die Ausgabe auf die Infos zum Superblock.

Optionen und Beispiele

```
Aufruf: mkfs.ext2 [-c|-l Dateiname] [-b Blockgröße] [-C Clustergröße]
        [-i Bytes-pro-Inode] [-I Inodegrösse] [-J Journal-Optionen]
        [-G Größe_der_Metagruppe] [-N Anzahl_der_Inodes]
        [-m Reservierte-Blöcke-Prozent] [-o Erzeuger-OS]
        [-g Blöcke-pro-Gruppe] [-L Volume-Label]
        [-M letztes-eingehängtes-Verzeichnis] [-O Eigenschaft[,...]]
        [-r fs-Revision] [-E erweiterte-Option[,...]]
        [-T Dateisystemtyp] [-U UUID] [-jnqvFKSV] Gerät [Block-Anzahl]
Dateisystemtyp = {floppy | small | big | huge | default}

mke2fs -t ext3 -T default /dev/sda4
dumpe2fs -h /dev/sda4
```

tune2fs

Wichtige Schalter **-i**: *interval-between-checks*, **-c**: *max-mount-counts* zwischen den Überprüfungen. Mit **-C** kann der Einhängecounter auf einen beliebigen Wert gesetzt werden.

```
Aufruf: tune2fs [-c max-Anzahl-Einhängen] [-e Fehler-Verhalten] [-g Gruppe]
        [-i Intervall[d|m|w]] [-j] [-J Journal-Optionen] [-l]
        [-m reservierte_Blöcke_Prozent] [-o [^]Einhäng_Optionen[,...]] [-p mmp_update_intervall]
        [-r Anzahl_reservierte_Blöcke] [-u Benutzer] [-C Anzahl_Einhängen]
        [-L Volume_Label] [-M letztes_eingehängtes_Verzeichnis]
        [-O [^]Eigenschaft[,...]] [-E erweiterte-Option[,...]]
        [-T letzter_Prüfzeitpunkt] [-U UUID] [-I neue_Inodegrösse] Gerät
```

mkswap

```
Usage:
 mkswap [options] device [size]

Options:
 -c, --check              check bad blocks before creating the swap area
 -f, --force              allow swap size area be larger than device
 -p, --pagesize SIZE      specify page size in bytes
 -L, --label LABEL        specify label
 -v, --swapversion NUM    specify swap-space version number
 -U, --uuid UUID          specify the uuid to use
 -V, --version            output version information and exit
 -h, --help               display this help and exit

mkswap /dev/sda4
```

swapon, swapoff

Partition als Swap-Bereich freigeben/sperren.

```
Usage:
 swapon [options] [<spec>]
 swapoff [options] [<spec>]

Options: //fuer swapon UND swapoff
 -a, --all               enable all swaps from /etc/fstab
 -h, --help              display help and exit
 -v, --verbose           verbose mode
 -V, --version           display version and exit

//ab hier nur Optionen für swapon
 -d, --discard           discard freed pages before they are reused
 -e, --ifexists          silently skip devices that do not exis
 -f, --fixpgsz           reinitialize the swap space if necessary
 -p, --priority <prio>   specify the priority of the swap device.
 -s, --summary           display summary about used swap devices and exit

//der Rest gilt wieder fuer beide
The <spec> parameter:
 -L <label>              LABEL of device to be used
 -U <uuid>               UUID of device to be used
 LABEL=<label>           LABEL of device to be used
 UUID=<uuid>             UUID of device to be used
 <device>                name of device to be used
 <file>                  name of file to be used

swapon -p 16384 /dev/sda4
swapon -s
swapoff /dev/sda4
```

xfs

- xfs ist das älteste Journaling-FS
- entwickelt von SGI, sehr stabil
- seit 2001 OpenSource.

```
mkfs -t xfs -f /dev/sda4
xfs_check /dev/sda4
mount /dev/sda4 /mnt
xfs_info /mnt
```

203.3 Creating and configuring filesystem options

Weight: 2

Description: Candidates should be able to configure automount filesystems using AutoFS. This objective includes configuring automount for network and device filesystems. Also included is creating filesystems for devices such as CD-ROMs.

Key Knowledge Areas

autofs configuration files UDF and ISO9660 tools and utilities awareness of CD-ROM filesystems (UDF, ISO9660, HFS) awareness of CD-ROM filesystem extensions (Joliet, Rock Ridge, El Torito)

Terms and Utilities

```
/etc/auto.master
/etc/auto.[dir]
mkisofs
dd
mke2fs
```

mkisofs

`mkisofs` = `genisoimage`

ISO9660 soll von Universal Disk Format abgelöst werden.

ISO9660 8+3 Zeichen - Dateinamen, 8 Verzeichnisebenen, 1 Sektor = 2kB, Dateigrösse = 4GB-1Sektor

ISO9660 Level 2 31 Zeichen - Dateinamen

ISO9660 Level 3 Dateien > 4GB

Rockridge Erweiterung des ISO9960 Standards; Extensions im Standard vorgesehen.

Joliet ganz anderes Dateisystem von M$. Nutzt ISO9960-Extensions **nicht**. D.H. baut eigenen Dateisystembaum auf.

ISO9660:1999 mehr als 8 Verzeichnisebenen, Punkt im Dateinamen hat keine Sonderbedeutung mehr, lange Dateinamen mit beliebigen Zeichen (207 / 221 Oktette)

203.4 udev Device Management

Weight: 1

Description: Candidates should understand device detection and management using udev. This objective includes troubleshooting udev rules.

Key Knowledge Areas

udev rules Kernel interface

Terms and Utilities

```
udevmonitor
/etc/udev
```

`udev` verwaltet die Gerätedateien in `/dev`. Die Verwaltung erfolgt *regelbasiert*.

Die Regeldateien stehen in

```
    /etc/udev/rules.d/
    /lib/udev/rules.d/

Beispieldatei: 70-persistent-net.rules

# PCI device 0x10ec:0x8168 (r8169)
SUBSYSTEM=="net", ACTION=="add", \
                  DRIVERS=="?*", \
                  ATTR{address}=="10:bf:48:e1:ed:e5", \
                  ATTR{dev_id}=="0x0", \
                  ATTR{type}=="1", \
                  KERNEL=="eth*", \
                  NAME="eth0"
```

Regeldateien *müssen* auf `.rules` enden. Regelsyntax:

== test auf Gleichheit

!= test auf Ungleichheit

= Wert an Schlüssel zuweisen

+= Wert zur Liste eines Schlüssels hinzufügen

:= Endgültige Zuweisung, spätere Änderung nicht möglich.

Schlüssel:

ACTION vergleiche mit Namen der Ereignis-Aktion

KERNEL vergleiche mit Namen des Ereignis-Geräts

ATTR{filename} vergleiche mit sysfs-Attribut-Werten

NAME Name einer Netzwerkschnittstelle

SYMLINK unter welchem Namen das Gerät unter /dev auftauchen soll

Ablauf:

- ```
  udevadm monitor
  ```

  starten.

- USB-Stick einstecken, Meldungen beobachten:

  ```
  KERNEL[23126.930524] add      /devices/pci0000:00/0000:00:1a.0/usb1/1-1/1-1.2 (usb)
  KERNEL[23126.930745] add      /devices/pci0000:00/0000:00:1a.0/usb1/1-1/1-1.2/1-1.2:1.0 (usb)
  KERNEL[23126.930851] add      /devices/pci0000:00/0000:00:1a.0/usb1/1-1/1-1.2/1-1.2:1.0/host6 (scsi)
  ```

- Pfad aus Meldung rauskopieren und mit `udevadm info` Attribute anschauen:
  ```

```
udevadm info --attribute-walk \
--path=/devices/pci0000:00/0000:00:1a.0/usb1/1-1/1-1.1/1-1.1:1.0/\
       host13/target13:0:0/13:0:0:0/scsi_device/13:0:0:0

looking at device '/devices/pci0000:00/0000:00:1a.0/usb1/1-1/1-1.1/1-1.1:1.0/\
       host13/target13:0:0/13:0:0:0/scsi_device/13:0:0:0':
    KERNEL=="13:0:0:0"
    SUBSYSTEM=="scsi_device"
    DRIVER==""

  looking at parent device '/devices/pci0000:00/0000:00:1a.0/\
       usb1/1-1/1-1.1/1-1.1:1.0/host13/target13:0:0/13:0:0:0':
    KERNELS=="13:0:0:0"
    SUBSYSTEMS=="scsi"
    DRIVERS=="sd"
    ATTRS{device_blocked}=="0"
    ATTRS{type}=="0"
    ATTRS{scsi_level}=="3"
    ATTRS{vendor}=="        "
    ATTRS{model}=="Da bin ich!     "
    ATTRS{rev}=="8.07"
    ATTRS{state}=="running"
    ATTRS{timeout}=="30"
    ATTRS{iocounterbits}=="32"
    ATTRS{iorequest_cnt}=="0x1ed"
    ATTRS{iodone_cnt}=="0x1ed"
    ATTRS{ioerr_cnt}=="0x2"
    ATTRS{evt_media_change}=="0"
    ATTRS{dh_state}=="detached"
    ATTRS{queue_depth}=="1"
    ATTRS{queue_type}=="none"
    ATTRS{max_sectors}=="240"
```

- Regel schreiben (**EINE** Zeile pro Regel, keine Newlines innerhalb einer Regel!!):

```
SUBSYSTEMS=="scsi", ACTION=="add", DRIVERS=="sd", ATTRS{rev}=="8.07", ATTRS{vendor}=="        ", \
                        ATTRS{model}=="Da bin ich!     ", SYMLINK+="spaceloop"
SUBSYSTEMS=="scsi", ACTION=="add", DRIVERS=="sd", ATTRS{vendor}=="Corsair ", \
                        ATTRS{model}=="Voyager         ", ATTRS{rev}=="3000", SYMLINK+="corsair"
```

- Regel einlesen:

```
udevadm control --reload-rules
```

204 Advanced Storage Device Administration

204.1 Configuring RAID

Weight: 2 Description: Candidates should be able to configure and implement software RAID. This objective includes using and configuring RAID 0, 1 and 5.

Key Knowledge Areas

Software raid configuration files and utilities

Terms and Utilities

```
mdadm.conf
mdadm
/proc/mdstat
fdisk
```

mdadm

MultiDiskAdministration

```
mdadm --query /dev/<laufwerk>  //--query = -Q

mdadm --create /dev/md0 --level=1 --raid-devices=2
       /dev/hd[ac]1    //erzeugt raid-1 aus /dev/hda1 und /dev/hdc1
                       //--level = -l
                       //--create = -C
                       //--raid-devices = n

mdadm --fail /dev/md0 /dev/sdb1    // /dev/sdb1 als fehlerhaft markieren
                                   //--fail = -f

mdadm --remove /dev/md0 /dev/sdb1  // /dev/sdb1 muss inaktiv sein
                                   // entweder spare oder als fail markiert
                                   // --remove = -r

mdadm --add /dev/md0 /dev/sdb1  // --add = a

mdadm -S /dev/md0  // -S = --stop

mdadm -A /dev/md0  // -A = --assemble
```

```
mdadm --create /dev/md0 /dev/sda3 /dev/sdb1 --level=1 --raid-devices=2

mdadm.conf:

DEVICE /dev/sda3
DEVICE /dev/sdb1
ARRAY /dev/md0 devices=/dev/sda3,/dev/sdb1

mit obiger mdadm.conf funktioniert

mdadm -A /dev/md0
```

204.2 Adjusting Storage Device Access

Weight: 1

Description: Candidates should be able to configure kernel options to support various drives. This objective includes software tools to view & modify hard disk settings.

Key Knowledge Areas

Tools and utilities to configure DMA for IDE devices including ATAPI and SATA Tools and utilities to manipulate or analyse system resources (e.g. interrupts) Awareness of sdparm command and its uses

Terms and Utilities

```
    hdparm
    sdparm
    tune2fs
    sysctl
    /dev/hd*
    /dev/sd*
```

204.3 Logical Volume Manager

Weight: 3

Description: Candidates should be able to create and remove logical volumes, volume groups, and physical volumes. This objective includes snapshots and resizing logical volumes.

Key Knowledge Areas

Tools in the LVM suite Resizing, renaming, creating, and removing logical volumes, volume groups, and physical volumes

Terms and Utilities

```
/sbin/pv*
/sbin/lv*
/sbin/vg*
mount
/dev/mapper/
```

Der LVM bildet physikalische Laufwerke (physical Volumes) auf logische Laufwerke ab (logical Volumes).

Physical Volume (PV) je ein PV wird aus *einer* physikalische Partition (Partitionstyp 82), einer kompletten Platte ohne Partitionstabelle oder einem anderen blockorientierten Gerät gebildet

Volume Group (VG) entspricht einem *logischen Laufwerk*. Eine VG wird aus PVs gebildet. Ist der Speicherplatz einer VG nicht mehr ausreichend, kann sie um weitere PVs erweitert werden.

Logical Volume (LV) entspricht einer *logischen Partition*, also einer Partition eines logischen Laufwerks. Ein LV kann über mehrere Festplatten verteilt sein.

Physical Extend Speichereinheit; vergleichbar mit Sektor einer physikalischen Partition; Defaultwert = 4MB

Beispiel: LVM auf alter OSX-Platte einrichten:

- alte Platte platt machen (gparted und fdisk haben es nicht geschafft, dd bringts):

```
dd if=/dev/zero of=/dev/sdb bs=512 count=1
```

- Physical Volume erzeugen. Hierbei werden die gewünschten Blockgeräte dem LVM bekannt gemacht.

```
pvcreate /dev/sdb
pvcreate /dev/sdb /dev/sda1 /dev/sda5
```

- PV-Search um das Ergebnis zu betrachten:

```
pvs -v
   Scanning for physical volume names
   Wiping cache of LVM-capable devices
 PV         VG   Fmt  Attr PSize   PFree   DevSize PV UUID
 /dev/sdb        lvm2 a-   152,67g 152,67g 152,67g DRnWxd-mED...
```

- Volume Group erzeugen:

```
   vgcreate volGroup /dev/sdb
      Volume group "volGroup" successfully created

   //bei mehreren PVs
   vgcreate volGroup /dev/sdb /dev/sda1 /dev/sda5
```

- VG-Search um das Ergebnis zu betrachten:

```
 vgs -v
    Finding all volume groups
    Finding volume group "volGroup"
  VG       Attr   Ext   #PV #LV #SN VSize   VFree   VG UUID
  volGroup wz--n- 4,00m   1   0   0 152,67g 152,67g ETINjh-Np...
```

- Noch detaillierter:

```
vgdisplay
  --- Volume group ---
  VG Name               volGroup
  System ID
  Format                lvm2
  Metadata Areas        1
  Metadata Sequence No  1
  VG Access             read/write
  VG Status             resizable
  MAX LV                0
  Cur LV                0
  Open LV               0
  Max PV                0
  Cur PV                1
  Act PV                1
  VG Size               152,67 GiB
  PE Size               4,00 MiB
  Total PE              39083
  Alloc PE / Size       0 / 0
  Free  PE / Size       39083 / 152,67 GiB
  VG UUID               ETINjh-Npkk-A1N1-KTQM-XLQP-Eee9-vJEqBT
```

- Volume Group um zusätzliches Laufwerk erweitern oder verkleinern:

```
   pvcreate /dev/sdc
   vgextend volGroup /dev/sdc
      Volume group "volGroup" successfully extended

   pvmove -v /dev/sdc  //extends wegschieben
      Finding volume group "volGroup"
      Archiving volume group "volGroup" metadata (seqno 7).
      Creating logical volume pvmove0
      No data to move for volGroup

   vgreduce volGroup /dev/sdc
      Removed "/dev/sdc" from volume group "volGroup"
```

- Logical Volume erzeugen

```
   lvcreate -L 150G -n fotos volGroup    //-n:namen des lv
   Logical volume "fotos" created
```

- Ergebnis überprüfen

```
   lvs -a -o +devices
   LV    VG       Attr   LSize   Origin ...  Devices
   fotos volGroup -wi-a- 150,00g             /dev/sdb(0)

   ls -l /dev/volGroup/
   insgesamt 0
   lrwxrwxrwx 1 root root 7 Sep 26 00:31 fotos -> ../dm-0
```

- Formatieren und mounten

```
mke2fs -t ext4 -T default /dev/volGroup/fotos
mount /dev/volGroup/fotos /mnt
df -h
   Dateisystem               Größe Benutzt Verf. Verw% Mount-PT
   ...
   /dev/mapper/volGroup-fotos  150G    2,4G  140G    2% /mnt
```

- Logical Volume erweitern. Mit der Option -L wird die Grösse angegeben (absolut oder incrementell). Nimmt man -l, ist die Grössenangabe in Vielfachen des PE. Verkleinern geht auch.

```
lvextend -L 160G /dev/volGroup/fotos
   Extending logical volume fotos to 160,00 GiB
   Logical volume fotos successfully resized

lvreduce -L -10G /dev/volGroup/fotos
   WARNING: Reducing active and open logical volume to 150,00 GiB
   THIS MAY DESTROY YOUR DATA (filesystem etc.)
   Do you really want to reduce fotos? [y/n]: y
   Reducing logical volume fotos to 150,00 GiB
   Logical volume fotos successfully resized

lvextend -L +10G /dev/volGroup/fotos

resize2fs /dev/volGroup/fotos
```

205 Networking Configuration

205.1 Basic networking configuration

Weight: 3

Description: Candidates should be able to configure a network device to be able to connect to a local, wired or wireless, and a wide-area network. This objective includes being able to communicate between various subnets within a single network.

Key Knowledge Areas

Utilities to configure and manipulate ethernet network interfaces Configuring wireless networks

Terms and Utilities

```
    /sbin/route
    /sbin/ifconfig
    /sbin/ip
    /usr/sbin/arp
    /sbin/iwconfig
    /sbin/iwlist
```

Beispiele:

```
route     //zeige routing-tabelle
route -n  //keine dns-lookups bei der ausgabe
route add default gw 192.168.178.1
route add -net 129.56.76.0 netmask 255.255.255.0 dev eth0
route add -net 129.57.66.0 netmask 255.255.255.0 gw ipx4
route del -net 129.57.66.0 netmask 255.255.255.0

ifconfig eth0 mtu 1300
ifconfig eth0 netmask 255.255.255.240 129.56.76.0 up

ifconfig eth0 inet6 add fe80::021b:63ff:fe9f:a23c

ip link show dev eth0
ip link set down dev eth0
ip link set up dev eth0

ip addr add 10.16.10.0/24 dev eth0
ip addr show dev eth0
ip -6 addr show eth0

ip addr add 2001:7c0:e100:1::1/64 dev eth0
ip -familiy inet6 addr add 2001:7c0:e100:1::1/64 dev eth0
ip -f inet6 addr add 2001:7c0:e100:1::1/64 dev eth0
ip -6 addr add 2001:7c0:e100:1::1/64 dev eth0
ip -6 addr del 2001:7c0:e100:1::1/64 dev eth0

ip route add 141.31.147.112/29 via 192.168.178.1
ip -6 route add 2001:7c0:e100:aaaa::/64 via 2001:7c0:e100:1::ffff

ip neigh show dev eth0 (list statt show geht auch)
ip neigh flush all
ip neigh flush 192.168.178.150
ip -6 neigh show

ip route show          (list statt show geht auch)

arp -an
arp -d 192.168.178.150
```

205.2 Advanced Network Configuration and Troubleshooting

Weight: 4

Description: Candidates should be able to configure a network device to implement various network authentication schemes. This objective includes configuring a multi-homed network device, configuring a VPN client and resolving communication problems.

Key Knowledge Areas

Utilities to manipulate routing tables Utilities to configure and manipulate ethernet network interfaces Utilities to analyse the status of the network devices Utilities to monitor and analyse the TCP/IP traffic OpenVPN

Terms and Utilities

```
/sbin/route
/sbin/ifconfig
/bin/netstat
/bin/ping
/usr/sbin/arp
/usr/sbin/tcpdump
/usr/sbin/lsof
/usr/bin/nc
/sbin/ip
/etc/openvpn/*
openvpn
nmap
wireshark
```

lsof

```
            lsof              //alle offenen dateien auflisten
            lsof -i -U        //-i internet, -U unix-sockets

            lsof -i 4 -a -p 1234    //-a = verUNDung
                                    // ipv4 und port 1234
            lsof -i 6               // ipv6

      To list all open files for login name ``abe'', or user ID 1234, or
      process 456, or process 123, or process 789, use:

            lsof -p 456,123,789 -u 1234,abe   //-u = user/user-id
```

205.3 Troubleshooting network issues

Weight: 5

Description: Candidates should be able to identify and correct common network setup issues, to include knowledge of locations for basic configuration files and commands.

Key Knowledge Areas

Location and content of access restriction files Utilities to configure and manipulate ethernet network interfaces Utilities to manage routing tables Utilities to list network states. Utilities to gain information about the network configuration Methods of information about the recognised and used hardware devices System initialisation files and their contents (SysV init process)

Terms and Utilities

```
    /sbin/ifconfig //s.o.
    /sbin/route    //s.o.
    /bin/netstat
    /etc/network
    /etc/sysconfig/network-scripts/
    /var/log/syslog
    /var/log/messages
    /bin/ping
    /etc/resolv.conf
    /etc/hosts
    /etc/hosts.allow
    /etc/hosts.deny
    /etc/hostname
    /etc/HOSTNAME
    /bin/hostname
    /usr/sbin/traceroute
    /usr/bin/dig
    /bin/dmesg
    /usr/bin/host
```

/etc/network/interfaces

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
      address 192.168.178.200
      netmask 255.255.255.0
      gateway 192.168.178.1

auto eth1
iface eth1 inet dhcp
```

/etc/sysconfig/network-scripts/ifcfg-<name>

Unter RH stehen im Verzeichnis `/etc/sysconfig/network-scripts/` einzelne Skripte und Konfigs für jede Netzwerkschnittstelle:
Beispiel:

```
ifcfg-lo
ifcfg-eth0
ifcfg-eth1
ifup
ifdown
ifup-ipv6
ifdown-ipv6
```

Beispiel von `ifcfg-eth0` statisch:

```
DEVICE=eth0
BOOTPROTO=none // weitere Werte dhcp | none | bootp
ONBOOT=yes     // oder no
NETWORK=10.0.1.0
NETMASK=255.255.255.0
IPADDR=10.0.1.27
USERCTL=no     //yes bedeutet: auch non-root-user haben Kontrolle
```

Beispiel von `ifcfg-eth0` dhcp:

```
DEVICE=eth0
BOOTPROTO=dhcp
ONBOOT=yes
```

Weitere Felder in den Dateien können sein:

```
BROADCAST=<address>
GATEWAY=<address>
HWADDR=<MAC-address>  //macht Konfig unabh. vom Namen (eth0, eth1, ...)
MACADDR=<MAC-address>  //überschreibt HW-Adresse der Schnittstelle.
                       //Nicht zusammen mit HWADDR verwenden!
NETWORK=<address> //veraltet; wird berechnet
```

`/etc/hosts.allow /etc/hosts.deny`

Aufbau der Zeilen in den Dateien hosts.allow und hosts.deny:

Prozess-Liste : Client-Liste [: Shell-Kommando]

Prozess-Liste: eine Liste (mit Komma oder Leerzeichen-Trenner) mit einem oder mehreren

- Namen von Diensten:
- Port-Nummern
- Wildcards

Client-Liste: eine Liste (mit Komma oder Leerzeichen-Trenner) mit einem oder mehreren

- Hostnamen
- IP-Addressen
- Wildcards
  - ALL : passt immer
  - LOCAL: passt auf Hostnamen, die keinen Punkt enthalten
- Patterns
  - 131.155.72.0/255.255.254.0 oder 141.31.147.112/28
  - **131.155.** entspricht 131.155.x.x
  - **.wara.de** passt auf msv.wara.de, caponova.wara.de usw.

- * und ? : Wildcards; kann nicht mit den vorigen drei kombiniert werden

Regeln, nach denen die Dateien abgearbeitet werden:

1. hosts.allow wir vor hosts.deny abgearbeitet.
2. Wenn ein (Prozess,Client)-Paar auf einen Eintrag in hosts.allow passt, wird die weitere Suche abgebrochen und Zugriff **gewährt**.
3. Wenn ein (Prozess,Client)-Paar auf einen Eintrag in hosts.deny passt, wird die weitere Suche abgebrochen und Zugriff **verweigert**.
4. gibt es keinen Match, wird der Zugriff **gewährt**.

205.4 Notify users on system-related issues

Weight: 1

Description: Candidates should be able to notify the users about current issues related to the system.

Key Knowledge Areas

Automate communication with users through logon messages. Inform active users of system maintenance

Terms and Utilities

```
/etc/issue
/etc/issue.net
/etc/motd
wall
/sbin/shutdown
```

206 System Maintenance

206.1 Make and install programs from source

Weight: 4

Description: Candidates should be able to build and install an executable program from source. This objective includes being able to unpack a file of sources.

Key Knowledge Areas

Unpack source code using common compression and archive utilities. Understand basics of invoking make to compile programs. Apply parameters to a configure script. Know where sources are stored by default.

Terms and Utilities

```
    /usr/src/
    gunzip
    gzip
    bzip2
    tar
    configure
    make
    uname
    install
```

httpd-Schnellinstallation

```
wget http://artfiles.org/apache.org//httpd/httpd-2.4.3.tar.bz2
tar xjf httpd-2.4.3.tar.bz2
cd httpd-2.4.3/
mkdir /home/micha/httpd
./configure --prefix=/home/micha/httpd --enable-ssl
make
make install
cd ~/httpd/
./bin/apachectl start  //vorher in httpd.conf Listen-Port auf 8000
```

206.2 Backup operations

Weight: 3

Description: Candidates should be able to use system tools to back up important system data.

Key Knowledge Areas

Knowledge about directories that have to be include in backups Awareness of network backup solutions such as Amanda, Bacula and BackupPC Knowledge of the benefits and drawbacks of tapes, CDR, disk or other backup media Perform partial and manual backups. Verify the integrity of backup files. Partially or fully restore backups.

Terms and Utilities

```
    /bin/sh
    cpio
    dd
    tar
    /dev/st*
    /dev/nst*
    mt
    rsync
```

cpio

cpio -o archivieren = in die Standardausgabe **aus**geben = copy-out cpio -i extrahieren = von der Standardeingabe **ein**lesen = copy-in

Beispiele:

```
ls | cpio -ov > directory.cpio   //nur dateien archivieren
find . -print -depth | cpio -ov > tree.cpio   //baum archivieren
find . -print -depth | cpio -ov | bzip2 > tree.cpio.bz2   //dito mit komprimierung

cpio -iv < directory.cpio   //dateien ins lokale verz. extrahieren
cpio -idv < tree.cpio       //baum ins lokale verz. extrahieren
                            // -d benoetigte verz. erzeugen

cpio -it < tree.cpio

bunzip2 -c bla.cpio.bz2 | cpio -idv   // bunzip2 -c : nach stdout dekompr.
```

mt

Magnetic Tape. Syntax:

```
mt -f <device> operation [anzahl]
```

Device ist z.b. /dev/st* = SCSI-Tape oder /dev/nst* = SCSI-No-Rewind-Tape (nach jedem Schreibvorgang zurückspulen / nicht zurückspulen)
operation ist eine von diesen:

fsf / bsf Forward / Backward space *anzahl* files = um *anzahl* Dateien vorspulen.

fsr / bsr dito, aber um *anzahl* Records spulen.

asf Absolute space to file number *anzahl* = rewind + fsf *anzahl*

rewind selbstredend

207 Domain Name Server

207.1 Basic DNS server configuration

Weight: 2
Description: Candidates should be able to configure BIND to function as a caching-only DNS server. This objective includes the ability to convert older BIND configuration files to newer format, managing a running server and configuring logging.
Key Knowledge Areas
BIND 9.x configuration files, terms and utilities Defining the location of the BIND zone files in BIND configuration files Reloading modified configuration and zone files
Terms and Utilities

```
/etc/named.conf
/var/named/*
/usr/sbin/rndc
kill
```

207.2 Create and maintain DNS zones

Weight: 2

Description: Candidates should be able to create a zone file for a forward or reverse zone or root level server. This objective includes setting appropriate values for records, adding hosts in zones and adding zones to the DNS. A candidate should also be able to delegate zones to another DNS server.

Key Knowledge Areas

BIND 9 configuration files, terms and utilities Utilities to request information from the DNS server Layout, content and file location of the BIND zone files Various methods to add a new host in the zone files, including reverse zones

Terms and Utilities

```
/var/named/*
zone file syntax
resource record formats
dig
nslookup
host
```

named.conf

```
options {
        directory "/var/cache/bind";

        forwarders {
             129.143.2.10;
        };

        listen-on-v6 { any; };
};

controls {
      inet 127.0.0.1 allow { localhost; } keys { rndc-key; };
};
include "/etc/rndc.key"  //include, da named.conf fuer jeden lesbar
# inhalt von rndc.key:
#
#key "rndc-key" {
#         algorithm hmac-md5;
#         secret "5qHP2EyHe9iKoWC8N7/xsw==";
#};

// prime the server with knowledge of the root servers
zone "." {
        type hint;
        file "/etc/bind/db.root";
};

// be authoritative for the localhost forward and reverse zones, and for
// broadcast zones as per RFC 1912

zone "localhost" {
        type master;
        file "/etc/bind/db.local";
};

zone "127.in-addr.arpa" {
        type master;
        file "/etc/bind/db.127";
};

zone "0.in-addr.arpa" {
        type master;
        file "/etc/bind/db.0";
};

zone "255.in-addr.arpa" {
        type master;
        file "/etc/bind/db.255";
};
```

Master-Slave

Auszug der named.conf auf Slave:

```
zone "peichl.private" in {
        type slave;
        file "slaves/peichl.private";
        masters { 10.20.30.41; };
};
```

Auszug der named.conf auf Master:

```
zone "peichl.private" in {
        type master;
        file "master/peichl.private";
        allow-transfer { 10.20.30.40; };
};
```

Zonendelegation

Beispiel: Zonendatei für die Zone `private.` :

```
$TTL 2d
@ IN SOA server.private. root.server.private. (
      2012100101 3h 1h 1w 1d
)

private. IN NS ns.private.   ; nameserver ist ns.private.
ns       IN A  10.20.30.40

;subdomain delegieren
sub      IN NS ns.sub   ; nameserver ist ns.sub.private.

ns.sub   IN A  10.20.30.50
```

rndc

```
rndc-confgen -a -c /etc/bind/rndc.key
```

rndc.conf

```
key "<key-name>" {
        algorithm hmac-md5;
        secret "<key-value>";
};

options {
        default-server  localhost;
        default-key     "<key-name>";
};

server localhost {
        key  "<key-name>";
};
```

`/etc/hosts`

Aufbau der Datei:

```
IP_address canonical_hostname [aliases...]

EXAMPLE
       127.0.0.1       localhost
       192.168.1.10    foo.mydomain.org       foo
       192.168.1.13    bar.mydomain.org       bar
       146.82.138.7    master.debian.org      master
       209.237.226.90  www.opensource.org
```

nslookup

Ist Antwort ”Non-authoritative”, wurde sie aus einem Cache geliefert und eben nicht von einem autoritativen Nameserver.

```
nslookup msv.wara.de 129.143.2.10
Server:                129.143.2.10
Address:        129.143.2.10#53

Name:        msv.wara.de
Address: 141.31.147.114

nslookup msv.wara.de
Server:                127.0.0.1
Address:        127.0.0.1#53

Non-authoritative answer:
Name:        msv.wara.de
Address: 141.31.147.114
```

207.3 Securing a DNS server

Weight: 2

Description: Candidates should be able to configure a DNS server to run as a non-root user and run in a chroot jail. This objective includes secure exchange of data between DNS servers.

Key Knowledge Areas

BIND 9 configuration files Configuring BIND to run in a chroot jail Split configuration of BIND using the forwarders statement

Terms and Utilities

```
    /etc/named.conf
    /etc/passwd
    DNSSEC
    dnssec-keygen
```

chroot

```
mkdir -p /chroot/bind
cd /chroot
mknod dev/null
mknod dev
mknod dev/zero

named -u bind -t /chroot/bin -c /chroot/bind/etc/bind/named.conf
```

208 Web Services

208.1 Implementing a web server

Weight: 3 Description: Candidates should be able to install and configure a web server. This objective includes monitoring the server's load and performance, restricting client user access, configuring support for scripting languages as modules and setting up client user authentication. Also included is configuring server options to restrict usage of resources.

Key Knowledge Areas

Apache 2.x configuration files, terms and utilities Apache log files configuration and content Access restriction methods and files mod_perl and PHP configuration Client user authentication files and utilities Configuration of maximum requests, minimum and maximim servers and clients

Terms and Utilities

```
access logs
error logs
.htaccess
httpd.conf
mod_auth
htpasswd
htgroup
apache2ctl
httpd
```

208.1.1 Apache-Direktiven

- Direktiven sind Schlüsselwörter in der Konfigurationsdatei
- Direktiven sind case-**in**sensitive
- Apache kennt mehrere 100 Direktiven
- Die Config-Dateien enthalten nur **eine** Direktive pro Zeile
- zwei verschiedene Arten von Direktiven:
    1. einfache Direktiven, genau eine Direktive pro Zeile; diese gelten für den gesamten Server

2. Tag-Direktiven, die xml-ähnlich weitere Direktiven einschliessen; die eingeschlossenen Direktiven gelten nur für die Bereiche, die mit den Tag-Direktiven definiert werden. Einige Beispiele (unvollst.):

   Directory
   Location
   Files
   VirtualHost

Directory

`Directory` kann andere Direktiven entalten, die dann nur für den angegebenen Pfad gelten im *lokalen Dateisystem*. Der Pfad kann bash-Wildcards enthalten (* und ?) oder auch reguläre Ausdrücke. Diese werden mit ~ eingeleitet. Beispiele:

```
<Directory /usr/local/httpd/htdocs/test>
   Options Indexes FollowSymLinks
</Directory>
<Directory /home/*/public_html>  ... </Directory>
<Directory ~ "^/www/.*/[0-9]{3}">  ... </Directory>
```

Files

wie `Directory`, gilt aber für einzelne Dateien. Macht natürlich mehr Sinn mit regulären Ausdrücken. Da nimmt man aber besser gleich **FilesMatch**.

```
<Files ~ "\.(gif|jpe?g|png)$"> %$

noch eins:

<Files ~ "^\.ht">
    Order allow,deny
    Deny from all
    Satisfy all
</Files>
```

Annm: '?' heisst 0 oder 1- mal das voranstehende Zeichen, '$' steht für das Stringende.

Location

`location` funktioniert sehr ähnlich zu `directory` (enthält auch Direktiven, erlaubt Wildcards und reguläre Ausdrücke).

Im Unterschied zur directory-Direktive, wird hier nicht der Dateipfad aus der *Sicht des Betriebssystems*, sondern ein Pfad oder eine URL angegeben. Dabei gilt der Pfad aus der *Sicht des Webservers*.

Syntax und Beispiele:

```
<Location URL-Pfad|URL> ... </Location>
<Location /test> ... </Location>
//vgl. mit oben.
//Doc-Root in beiden Faellen: /usr/local/httpd/htdocs/
```

208.1.2 Globale Konfiguration

Die wichtigsten globalen Direktiven an einer Beispiel-Konfig. In aktuellen Distros wird die Konfig auf mehrere Dateien verteilt.

```
ServerRoot "/etc/apache2"

#das besser in sites-available/default eintragen
#ServerName raspiMicha.fritz.box

# The number of seconds before receives and sends time out.
Timeout 300

# More than one request per tcp-connection on/off.
KeepAlive On

# Maximum number of requests per tcp-connection. 0 = unlimited.
MaxKeepAliveRequests 100

# tcp-connection erst nach 5s abbauen
KeepAliveTimeout 5

# Initial number of server processes to start
StartServers          5

# selbsterklaerend
MinSpareServers       5
MaxSpareServers      10

# Maximum number of simultaneous client connections
MaxClients          150

# Maximale zahl von tcp-verbindungsaufbauten, bevor prozess stirbt. 0 = unlimited
MaxRequestsPerChild   0


# kann auch ueber variablen in /etc/apache2/envvars gesetzt werden
User www-data
Group www-data
LockFile /var/lock/apache2/accept.lock
PidFile /var/run/apache2.pid
ErrorLog /var/log/apache2/error.log

AccessFileName .htaccess

<Files ~ "^\.ht">
    Order allow,deny
    Deny from all
    Satisfy all
</Files>

# DefaultType is the default MIME type the server will use for a document
DefaultType None

LogLevel warn

# Include module configuration:
Include mods-enabled/*.load
Include mods-enabled/*.conf

# ports to listen on
Listen 80

#
# The following directives define some format nicknames for use with
# a CustomLog directive (see below).
# If you are behind a reverse proxy, you might want to change %h into %{X-Forwarded-For}i
#
LogFormat "%v:%p %h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\"" vhost_combined
LogFormat "%h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %O" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent

NameVirtualHost *:80
# Include the virtual host configurations:
Include sites-enabled/
```

sites-enabled enthält diese Standardkonfiguration:

```
<VirtualHost *:80>
       ServerName raspiMicha.fritz.box
       ServerAdmin webmaster@localhost

        DocumentRoot /var/www
        <Directory />
                Options FollowSymLinks
                AllowOverride None
        </Directory>
        <Directory /var/www/>
                Options Indexes FollowSymLinks MultiViews
                AllowOverride None
                Order allow,deny
                allow from all
        </Directory>

        ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
        <Directory "/usr/lib/cgi-bin">
                AllowOverride None
                Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
                Order allow,deny
                Allow from all
        </Directory>

        ErrorLog ${APACHE_LOG_DIR}/error.log

        # Possible values include: debug, info, notice, warn, error, crit,
        # alert, emerg.
        LogLevel warn

        CustomLog ${APACHE_LOG_DIR}/access.log combined
</VirtualHost>
```

In diesem Beispiel legt die Direktive `Options` fest, welche Eigenschaften im konfigurierten Verzeichnis (`Directory`) zur Verfügung stehen:

Indexes wenn das Verzeichnis *nicht* die unter `DirectoryIndex` angegebene Datei enthält (z.B. `index.html`), sorgt `mod_autoindex` für eine Auflistung des Verzeichnisinhalts.

FollowSymLinks, SymLinksIfOwnerMatch selbsterklärend

ExecCGI selbsterklärend

Die Zeichen + oder - vor den Optionen haben folgende Funktion:

Sind durch die Konfiguration mehrere Optionen innerhalb eines Verzeichnisses wirksam, wird die Option angewendet, die am weitesten unten in einem Unterverzeichnis deklariert wird.

Beispiel:

```
<Directory /web/docs>
Options Indexes FollowSymLinks
</Directory>

<Directory /web/docs/spec>
Options Includes
</Directory>
```

in `/web/docs/spec` wirkt **nur** die Option `Includes`.
Mit den + und - Zeichen kann man nun Optionen im Unterverzeichnis *hinzufügen* oder *wegnehmen*:

```
<Directory /web/docs>
Options Indexes FollowSymLinks
</Directory>

<Directory /web/docs/spec>
Options +Includes -Indexes
</Directory>
```

Ergebnis: im Verzeichnis /web/docs/spec wirken jetzt die Optionen `FollowSymLinks` **und** `Includes`

208.1.3 Nur bestimmte Client-Adressen zulassen

Direktiven: `Allow from, Deny from, Order`
Beispiele:

```
Allow from apache.org
Allow from .net example.edu
Allow from 10.1.2.3
Allow from 192.168.1.104 192.168.1.205
Allow from 10.1
Allow from 10 172.20 192.168.2
Allow from 10.1.0.0/255.255.0.0
Allow from 10.1.0.0/16
Allow from 2001:db8::a00:20ff:fea7:ccea
Allow from 2001:db8::a00:20ff:fea7:ccea/10
```

Direktive `Order`

Allow,Deny der Zugriff wird nur **gewährt**, wenn mindestens eine Allow- und keine Deny-Regel passt. Sonst wird immer verweigert.

Deny,Allow der Zugriff wird nur **verweigert**, wenn mindestens eine Deny- und keine Allow-Regel passt. Sonst wird immer zugelassen.

208.1.4 Alias und Weiterleitung

Mit der Direktive `Alias` kann man Inhalt an beliebigen Stellen im Dateisystem speichern, nicht mehr nur unter `DocumentRoot`.

Kein /-Zeichen am Pfadende!!!!

Entsprechendes geht auch mit `AliasMatch` und RegExp.

```
Alias /image /ftp/pub/image
<Directory /ftp/pub/image>
  Order allow,deny
  Allow from all
</Directory>
```

`/image` im ist ein URL-Pfad.
Mit `Redirect` kann man auf eine andere URL verweisen:

```
Redirect /service http://foo2.example.com/service
```

Jeder Request, der mit dem URL-Pfad /service beginnt wird weitergeleitet:
`http://example.com/service/foo.txt`
⇒
`http://foo2.example.com/service/foo.txt`

208.1.5 .htaccess: hypertext access

Mit speziellen Dateien, die in der Standard-Konfiguration `.htaccess` heissen und im Dateisystem des Web-Contents verteilt werden, kann man die Serverkonfiguration dezentralisiert vornehmen. Stichwort: *directory-level configuration*.

- .htaccess-Dateien verwenden dieselbe Syntax wie `apache2.conf`
- .htaccess-Dateien werden meistens dafür verwendet, um Zugriff auf bestimmte Verzeichnisse mit Passwortschutz zu versehen.
- ob eine bestimmte Direktive in .htaccess-Dateien erlaubt ist, muss man in deren Doku nachschlagen: unter dem Stichwort "**Context**" steht zu jeder Direktive, wo sie *erlaubt* ist.
- ob aber eine Direktive in einer .htaccess-Datei *ausgewertet* wird, wird mit der Direktive `AllowOverride` in `apache2.conf` festgelegt. Diese muss mindestens die Kategorie enthalten, die in der Doku einer Direktive unter dem Stichwort "**Override**" angegeben ist.

Beispiel `AuthType`, die Doku sagt:

```
  Description:  Type of user authentication
  Syntax:       AuthType Basic|Digest
  Context:      directory, .htaccess
  Override:     AuthConfig
```

Und entsprechend sieht die apache2.conf aus:

```
<Directory /home/*/public_html>
   AllowOverride FileInfo AuthConfig Limit Indexes
   Options MultiViews Indexes SymLinksIfOwnerMatch IncludesNoExec
    ... gekuerzt ...
</Directory>
```

Die Verwendung von .htaccess sollte vermieden werden. Besser, alles was in .htaccess steht in einen Directory-Block in httpd.conf schreiben.

Häufig verwendete Direktiven zu `.htaccess`:

Diese Direktiven dürfen natürlich auch in einem Directory-Block in `apache2.conf` stehen:

LoadModule Wird .htaccess für *Basic*-Authorisierung verwendet, müssen ein paar Module geladen werden: `mod_auth_basic`, `mod_authn_file` und `mod_authz_user`. Syntax:

```
LoadModule auth_basic_module /usr/lib/apache2/modules/mod_auth_basic.so
LoadModule authn_file_module /usr/lib/apache2/modules/mod_authn_file.so
LoadModule authz_user_module /usr/lib/apache2/modules/mod_authz_user.so
```

Für *Digest*-Authorisierung siehe unten.

AccessFileName Name der dezentralen Konfigurationsdatei. Defaultwert: `.htaccess`

AuthType *Basic* oder *Digest*, *Basic* heiss Klartextübertragung des Passworts, *Digest* können nur moderne Browser

AuthName Name des Realms = geschützter Bereich. Wir meistens vom Browser angezeigt.

AuthUserFile Name der Textdatei, die die Benutzernamen und Passwörter für den Realm enthält. Dateiaufbau:

```
student:QPGfcIZVs1QfI
konfi:.9cHqkOlhSsVM
boom:8xc7BDJX7g1ks
watz:q3YxZa1AkXXDc
```

AuthGroupFile Name der Textdatei, die die Namen einer Benutzergruppe mit ihren Mitgliedern enthält. Beispiel für den Dateiaufbau:

```
lehrer: micha peter josef
schueler: kevin finn vanessa linn
```

Require listet die erlaubten User auf. Es gibt:

user alle aufgelisteten Benutzer sind erlaubt

group alle Benutzer dieser Gruppe (Gruppe in *AuthGroupFile* definiert) sind erlaubt

valid-user alle Benutzer, die in *AuthUserFile* und in *AuthGroupFile* eingetragen sind, sind erlaubt

Beispiel für eine dezentrale `.htaccess`-Datei:

```
AuthType Basic
AuthName "waraacad"
AuthUserFile /home/dienert/.htpasswd
AuthGroupFile /home/dienert.htgroup
Require user student | Require group lehrer schueler
```

Beispiel für Verwendung der Direktiven direkt in `apache2.conf`:

```
<Directory /var/www/privat>
   AuthType Basic
   AuthName "privat"
   AuthUserFile /etc/apache2/htpasswd/passwdTux
   Require user tux
</Directory>
#analog mit Location-Direktive
<Location /privat>
   AuthType Basic
   AuthName "privat"
   AuthUserFile /etc/apache2/htpasswd/passwdTux
   Require user tux
</Location>
```

Für `AuthType Digest` müssen 2 Module geladen werden. Dazu (Debian,Ubuntu) in `mods-enabled` zwei Symlinks erzeugen und die Passwortdatei mit dem Kommando `htdigest` generieren:

```
 auth_digest.load -> ../mods-available/auth_digest.load
 authn_file.load -> ../mods-available/authn_file.load
 htdigest -c ../htpasswd/passwdTuxDgst private tux

 allgemein:
 htdigest [ -c ] passwdfile realm username

 Laden eines Moduls mit Direktive:

 LoadModule auth_digest_module /usr/lib/apache2/modules/mod_auth_digest.so
 LoadModule authn_file_module /usr/lib/apache2/modules/mod_authn_file.so
```

Beispiel für den Teil in `apache2.conf` (ich weiss, normalerweise heisst es nicht `apache2.conf`, aber auf brauchbaren Distros schon ...):

```
<Directory /var/www/privat>
   AuthType Digest
   AuthDigestProvider file
   AuthName "private"
   AuthUserFile /etc/apache2/htpasswd/passwdTuxDgst
   Require user tux
</Directory>
```

208.1.6 Dynamische Webseiten

CGI

Common Gateway Interface: Apache kommuniziert mit dem CGI-Programm über Umgebungsvariablen (QUERY_STRING) oder Standardeingabe und liest aus der Standardausgabe des CGI-Programms.

1. Möglichkeit mit Direktive `ScriptAlias`: `ScriptAlias` funktioniert genau gleich wie die `Alias`-Direktive, aber Apache betrachtet alle Dateien im Zielverzeichnis als CGI-Programme (Siehe 208.1.4. Beispiel:

```
LoadModule cgi_module modules/mod_cgi.so ScriptAlias /cgi-bin/
/usr/local/apache2/cgi-bin/
```

2. Möglichkeit mit der Option `ExecCGI` und der Direktive `AddHandler` Beispiel um im `Userdir` CGI-Ausführung zu erlauben:

```
<Directory /home/*/public_html>
    Options +ExecCGI
    AddHandler cgi-script .cgi .pl
</Directory>
```

   Alle Dateien mit der Endung `.cgi` oder `.pl` werden als CGI-Programm betrachtet.

   Es ist auch möglich, **php**-Skripte via cgi aufzurufen. Das ist hier nicht beschrieben.

   Perl-Beispiel: Das Perl-Script muss z.B. `hello.pl` heissen und in `~/public_html` stehen. Das x-Rechte-Bit *muss* gesetzt sein!

```
#!/usr/bin/perl
print "Content-type: text/html\n\n";
print "jipee! <br/> perl funzt!<br/>";
print "zeit (in s seit 1970-01-01): ";
print time;
```

AddHandler und SetHandler

Die beiden Direktiven `AddHandler` und `SetHandler` sind sehr ähnlich. Beide verbinden Dateien mit einem Handler. Dabei ist ein Handler eine bestimmte *Aktion*, die Apache mit der Datei ausführt.

AddHandler Syntax: `AddHandler handler-name extension`
Alle Dateien, die die Dateiendung `extension` haben, werden von dem Handler `handler-name` ausgeliefert. Beispiel:

```
AddHandler cgi-script .cgi
```

SetHandler Syntax: `SetHandler handler-name | None` Bei `SetHandler` gibt es keine Datei-Extension, da `SetHandler` z.B. in einer `Location`-, `Directory`- oder `FilesMatch`-Umgebung stehen muss. Alle passenden Dateien werden dann mit dem Handler ausgeliefert.

```
<Location /machPerl>
   SetHandler perl-script
   PerlResponseHandler ModPerl::Registry
</Location>
```

Direkte Ausführung von Perl und PHP

PHP Modul laden. Evtl. muss vorher das Paket `libapache2-mod-php5` installiert werden:

```
LoadModule php5_module /usr/lib/apache2/modules/libphp5.so
```

PHP konfigurieren. Gezeigt ist der Teil, der benötigt wird. Bei einer Installation von PHP5 wird standardmässig das Ausführen von PHP in den Userdirs verboten. Dieser Konfigurationsteil muss auskommentiert werden. Der auskommentierte Teil ist hier nicht gezeigt.

```
<FilesMatch ".+\.ph(p[345]?|t|tml)$">%$
    SetHandler application/x-httpd-php
</FilesMatch>
```

```
 <!DOCTYPE HTML>
 <html>
   <head>
     <title> Hallo-Welt-Beispiel </title>
   </head>
   <body>
     <?php
        echo 'Hallo Welt!<br/>';
        echo 'date: ';
        @date_default_timezone_set("Europe/Berlin");
        echo date(DATE_RFC822);
        ?>
   </body>
 </html>
```

Perl Modul laden. Dazu muss vorher das Paket `libapache2-mod-perl2` installiert werden:

```
LoadModule perl_module /usr/lib/apache2/modules/mod_perl.so
```

Perl konfigurieren.

```
<Directory /home/*/public_html>
   AddHandler perl-script .pl
   PerlResponseHandler ModPerl::Registry
   PerlOptions +ParseHeaders
   Options +ExecCGI
   AddHandler cgi-script .cgi
</Directory>
```

208.2 Maintaining a web server

Weight: 2 Description: Candidates should be able to configure a web server to use virtual hosts, Secure Sockets Layer (SSL) and customise file access.

Key Knowledge Areas

SSL configuration files, tools and utilities SSL certificate handling Apache 2.x virtual host implementation (with and without dedicated IP addresses) Using redirect statements in Apache's configuration files to customise file access

Terms and Utilities

```
Apache2 configuration files
/etc/ssl/*
openssl
```

208.2.1 Virtual Hosts

Der Begriff *Virtual Hosts* bedeutet, mehrere Web-Auftritte auf einer physikalischen Maschine zu betreiben.

Man unterscheidet zwischen

IP-Based-Virtual Hosts
- IP-Adresse der TCP-Verbindung wird verwendet, um zu bestimmen, welcher virtuelle Host die Seiten liefern muss
- pro virtual Host eine IP-Adresse
- mehrere IP-Adressen auf dem physikalischen I/F der Maschine

Name-Based-Virtual Hosts
- Der Hostname im http-Request-Header wird verwendet, um den richtigen Virtual Host zu bestimmen
- einfacher zu konfigurieren: DNS-Einträge für jeden Web-Auftritt auf die eine IP-Adresse, Webserver so konfigurieren, dass er die unterschiedlichen Hostnamen erkennt
- Adressen werden gespart
- SSL auch möglich, wenn `mod_ssl` mit SNI-Support verwendet wird.
- zu bevorzugende Variante!

Name-Based-Virtual Hosts

Fügt man zu einem bestehenden Server einen Virtual Host hinzu, *verschwindet* der ursprüngliche Hauptserver, wenn er nicht auch in einen `<VirtualHost>`-Block gestellt wird

```
NameVirtualHost *:80

<VirtualHost *:80>
   ServerName mmm.alfred.org
   ServerAlias alfred.org alfred.mad *.alfred.mad
   DocumentRoot /home/micha/alfred
   <Directory /home/micha/alfred/>
      Options Indexes FollowSymLinks
      Order allow,deny
      allow from all
   </Directory>
</VirtualHost>
```

Damit das funktioniert, müssen die Namen alfred.org alfred.mad mmm.alfred.org usw. natürlich aufgelöst werden. Zum Testen genügt ein Eintrag in in /etc/hosts auf der Maschine, von der aus man die Seiten aufruft:

```
192.168.178.36   mmm.alfred.org
192.168.178.36   alfred.org
192.168.178.36   alfred.mad
```

Direktiven:

NameVirtualHost Zwingend erforderlich für name-based Virtual Hosts.

Syntax:

```
NameVirtualHost addr[:port]
```

addr ist die Adresse, unter der alle name-based Virtual Hosts erreicht werden. Verwendet man statt der Adresse einen *, reagiert der Server auf Anfragen an *allen* phyiskalischen Interfaces.

<VirtualHost> erzeugt einen VirtualHost-Definitions-Block. Alle Direktiven, die darin eingeschlossen sind, gelten dann nur für diesen VirtualHost.

Syntax:

```
<VirtualHost addr[:port] [addr[:port]] ...> ... </VirtualHost>
```

Steht statt der IP-Adresse ein *, gilt jede IP-Adresse. Nur in Verbindung mit `NameVirtualHost` erlaubt!

ServerName, ServerAlias mit den Einträgen von ServerName und ServerAlias wird der Virtual Host eindeutig identifiziert.

Beim default VH = Hauptserver *muss* dann auch ein ServerName eingetragen werden. Dieser kann aber auch in der globalen Konfig stehen.

IP-Based Virtual Hosts

Beim IP-Based-Virtual-Hosting möchte Apache für jeden Virtual Host eine andere IP/Port-Kombination sehen. D.h. man muss

- entweder mehrere verschiedene IP-Adressen auf einem physikalischen Interface konfigurieren
- und / oder verschiedene Ports verwenden. Jeder Port muss mit der Direktive `Listen` abgehört werden.

```
Listen 8001
Listen 8002
```

Konfiguriert werden kann Apache danach auf zwei Arten:

1. für jeden Virtual Host eine eigene httpd-Installation mit eigner Konfiguration vornehmen und jeweils die `Listen` - Direktive mit der richtigen IP/Port-Kombination eintragen:

   ```
   Listen 10.20.30.40:80
   ```

2. die VirtualHost-Direktive verwenden:

   ```
   <VirtualHost 192.168.0.1:80>
   ServerAdmin webmaster@smallco.example.com
   DocumentRoot /groups/smallco/www
   ServerName smallco.example.com
   ErrorLog /groups/smallco/logs/error_log
   TransferLog /groups/smallco/logs/access_log
   </VirtualHost>

   <VirtualHost 192.168.0.2:80>
   ServerAdmin webmaster@baygroup.example.org
   DocumentRoot /groups/baygroup/www
   ServerName baygroup.example.com
   ErrorLog /groups/baygroup/logs/error_log
   TransferLog /groups/baygroup/logs/access_log
   </VirtualHost>
   ```

   Auch hier müssen wohl alle IP/Port-Kombinationen mit `Listen` abgehört werden.

208.2.2 SSL

Mit dem Modul mod_ssl bekommt Apache eine Schnittstelle zur OpenSSL-Bibliothek.

```
Listen 443
<VirtualHost _default_:443>
        ServerAdmin webmaster@localhost

        DocumentRoot /var/www
        <Directory />
                Options FollowSymLinks
                AllowOverride None
        </Directory>
        <Directory /var/www/>
                Options Indexes FollowSymLinks MultiViews
                AllowOverride None
                Order allow,deny
                allow from all
        </Directory>

        ScriptAlias /cgi-bin/ /usr/lib/cgi-bin/
        <Directory "/usr/lib/cgi-bin">
                AllowOverride None
                Options +ExecCGI -MultiViews +SymLinksIfOwnerMatch
                Order allow,deny
                Allow from all
        </Directory>

        ErrorLog ${APACHE_LOG_DIR}/error.log

        # Possible values include: debug, info, notice, warn, error, crit,
        # alert, emerg.
        LogLevel warn

        CustomLog ${APACHE_LOG_DIR}/ssl_access.log combined

        #   Enable/Disable SSL for this virtual host.
        SSLEngine on

        #   A self-signed (snakeoil) certificates
        SSLCertificateFile    /etc/ssl/certs/ssl-cert-snakeoil.pem
        SSLCertificateKeyFile /etc/ssl/private/ssl-cert-snakeoil.key

</VirtualHost>
```

208.3 Implementing a proxy server

Weight: 2 Description: Candidates should be able to install and configure a proxy server, including access policies, authentication and resource usage.

Key Knowledge Areas

Squid 2.x configuration files, terms and utilities Access restriction methods Client user authentication methods Layout and content of ACL in the Squid configuration files

Terms and Utilities

```
    squid.conf
    acl
    http_access
```

208.3.1 ACL

Syntax:

```
acl name typ argument
```

Wichtige Typen:

url_regex

src

dstdomain

proxy_auth

208.3.2 Filterregeln

Syntax:

```
http_access allow|deny [!]aclname1 [aclname2] [ ... ]
```

```
#acl debuggen
debug_options 28,3

#abhoeren aller IP-adressen des rechners
#http_port 8080

#nur auf eine adresse hoeren
http_port 192.168.178.36:8080

# keine einheit bei 512 MB angeben, ist immer MB!!
cache_dir ufs /var/spool/squid 512 16 256

cache_mem 256 MB


access_log /var/log/squid/access.log squid

auth_param basic program /usr/lib/squid/ncsa_auth /etc/squid/squid_passwd

#################################### acl ################
# acl name typ argument

# 'all' ist die liste, die fuer alle IP-adressen gilt:
acl all src 0.0.0.0/0.0.0.0
# acl all src all    muesste das selbe sein

acl localhost src 127.0.0.1/32

acl waelderstrasse7 src 192.168.178.0/255.255.255.0
acl leo src 192.168.178.200/255.255.255.255

# der punkt vor der domaene wirkt wie % in sql !!
# d.h. ohne punkt zaehlen nur exakte matches !!
acl facebook1 dstdomain .facebook.com
acl facebook2 url_regex facebook

# methode CONNECT steht fuer ssl-verbindungen
acl connect method CONNECT

acl authorisierteUser proxy_auth REQUIRED

#################################### http_access ################
# http_access allow|deny [!]aclname ...

# beim ersten match wird die liste nicht weitergeprueft
# bei 2 acls in einer regel, muessen beide gelten (logisch UND)
# die einzelnen Zeilen dagegen sind ODER verknuepft
# die regel fuer proxy_auth muss am anfang stehen(?).
http_access allow authorisierteUser
#
http_access deny connect facebook2
http_access allow leo
http_access allow localhost
http_access allow waelderstrasse7

# wenn keine access_regel passt, wird das GEGENTEIL
# der letzten regel angewendet. damit das nicht passiert
# schreiben wir hier deny all. die regel passt auf jeden fall
http_access deny all
```

209 File Sharing

209.1 SAMBA Server Configuration

Weight: 4 Description: Candidates should be able to set up a SAMBA server for various clients. This objective includes setting up Samba for login clients and setting up the workgroup in which a server participates and defining shared directories and printers. Also covered is a configuring a Linux client to use a Samba server. Troubleshooting installations is also tested.

Key Knowledge Areas

Samba 3 documentation Samba configuration files Samba tools and utilities Mounting Samba shares on Linux Samba daemons Mapping Windows usernames to Linux usernames User-Level and Share-Level security

Terms and Utilities

```
smbd, nmbd
smbstatus
testparm
smbpasswd
nmblookup
smbclient
net
/etc/smb/*
/var/log/samba/
```

209.1.1 smbd, nmbd

Eine Beispiel-Konfig:

```
[global]
        netbios name = RASPINETBIOSNAM
        workgroup = RASPIWORKGROUP
        server string = raspi als smb-server
        printcap name = CUPS
        printing = CUPS
        load printers = yes
        security = user
        encrypt passwords = yes
#         smb passwd file = /etc/samba/smbpasswd
#         passdb backend = tdbsam:/etc/samba/private/passdb.tdb
        passdb backend = smbpasswd:/etc/samba/smbpasswd
        username map = /etc/samba/user.map
#       aufbau der mapdatei
#       linuxuser = smbuser [smbuser ...]
#       micha = micha michael
        domain logons = yes
        logon script = login.bat
[public]
        comment = OeffentlicherNurLeseOrdner
        path = /home/public
        read only = yes
        guest ok = no
        # public = yes : synonym zu guest ok
        # write list funktioniert nur mit security = user
        # und natuerlich auch nur, wenn die unix-rechte
        # des verzeichnisses schreiben fuer alle   erlauben!
        write list = micha, alfred
[netlogon]
        comment = wintendoPCs erwarten dieses share
        # /home/netlogon muss loginScript und groupPolicy enthalten
        # beide als DOS-dateien speichern!
        path = /home/netlogon
        guest ok = yes
        writable = no
[homes]
        comment = Home Directories
        browseable = no
        read only = no
        create mask = 0775
        directory mask = 0775
        # %S wird durch 'servicename' d.h. 'username' ersetzt
        # wenn das home-dir so heisst wie der user
        valid users = %S
[printers]
        comment = Drucker
        path = /var/spool/samba
        browsable = no
        guest ok = no
        writeable = no
        printable = yes
```

Beschreibung:

netbios name Standard-Wert ist das erste Label des dns-Namens des Rechners, auf dem der Server läuft. Mit *netbios name*, kann man einen anderen Namen vergeben.

workgroup die Wintendo-Workgroup, zu der dieser Server gehört

server string beliebiger Text, der den Server beschreibt

printing was für ein Printing-Subsystem soll verwendet werden: CUPS/lprng

security **wichtigste Variable der Konfiguration(!)** Werte:

share Wintendo-Benutzer, die sich mit smbd verbinden müsssen nicht notwendigerweise Unix-Benutzer sein. Gut für Gast-Zugänge.

user beste Wahl für überschaubare Benutzergruppe. Benutzer, die sich verbinden, müssen sich gegenüber smbd authentifizieren (name/password) und smbd entscheidet, welchen access-level der Benutzer erhält.

domain smbd reicht Authentifizierung an M$-Domänenkontroller weiter. smbd agiert also nur als Gateway für den Authentifizierungsprozess.

encrypt passwords `/etc/samba/smbpasswd` enthält smb-User mit deren Passwörtern. Der Password-Algorithmus von Linux und Windows ist verschieden, deshalb ist eine extra smbpasswd-Datei notwendig. Zum Kommando `smbpasswd` siehe unten.

write list Liste mit Benutzern, die Schreibrechte haben. Das +-Zeichen zeigt eine Unix-Usergruppe an. Das -Zeichen eine NIS- **oder** Unix-Gruppe.

valid users Gültigkeit des shares auf eine Usergruppe einschränken. Verwendet man '%S' als User, wird %S durch den aktuellen Servicename ersetzt. Sehr nützlich bei den Home-Directories!

netlogon Die Dateien login.cmd und config.pol müssen im DOS-Textdateiformat sein:

```
sed 's/$/^M/' file
```

printers smb und cups sind inzwischen eng verzahnt. Damit cups mit smb funktioniert, muss samba gegen libcups gelinkt sein. Testen mit:

```
ldd /usr/sbin/smbd | grep -i cups
        libcups.so.2 => /usr/lib/arm-linux-gnueabihf/libcups.so.2 (0xb64cc000)
```

209.1.2 Samba als PDC

Maschinenaccount erzeugen:

```
adduser --force-badname --system wintendo$
smbpasswd -a -m wintendo$
```

Ausserdem muss das Linux-Verzeichnis /home/netlogon die Dateien

```
login.cmd
config.pol
```

enthalten. Beide mit DOS-Zeilenenden (s.o.).

209.1.3 testparm

Sehr hilfreich!!

```
root@raspiMicha:/etc/samba# testparm smb.conf
Load smb config files from smb.conf
rlimit_max: increasing rlimit_max (1024) to minimum Windows limit (16384)
Processing section "[public]"
Processing section "[netlogon]"
Processing section "[homes]"
Loaded services file OK.
Server role: ROLE_DOMAIN_PDC
Press enter to see a dump of your service definitions

[global]
        workgroup = RASPIWORKGROUP
        netbios name = RASPINETBIOSNAME
        server string = raspi als smb-server
        passdb backend = smbpasswd:/etc/samba/smbpasswd
        printcap name = CUPS
        logon script = login.bat
        domain logons = Yes
        idmap config * : backend = tdb

[public]
        comment = OeffentlicherNurLeseOrdner
        path = /home/public
        write list = +leerer, root

[netlogon]
        comment = wintendoPCs erwarten dieses share
        path = /home/netlogon
        guest ok = Yes

[homes]
        comment = Home Directories
        valid users = %S
        read only = No
        create mask = 0775
        directory mask = 0775
        browseable = No
```

209.1.4 smbpasswd

Samba kann neben der Text-Passwortdatei auch eine TrivialeDatenBank (tdb) für die Passwörter verwenden. Was verwendet werden soll und wo die Dateien stehen wird in smb.conf festgelegt:

```
#       alte variante:
#       smb passwd file = /etc/samba/smbpasswd

#       neue variante
        passdb backend = smbpasswd:/etc/samba/smbpasswd

#       version mit trivialdatenbank
#       passdb backend = tdbsam:/etc/samba/private/passdb.tdb
```

```
smbpasswd -a (add)
          -x (delete)
          -d (disable)
          -e (enable)    username
```

Inhalt von smbpasswd (user=micha):

```
micha:1001:XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX:407180ECBADB2BD37DF47E51DAD4D0E8:[U          ]:LCT-516DDE32:
```

209.1.5 smbstatus

```
root@raspberrypi:/etc/samba# smbstatus

Samba version 3.6.6
PID     Username      Group         Machine
-------------------------------------------------------------------
 <processes do not show up in anonymous mode>

Service      pid     machine       Connected at
-------------------------------------------------------
IPC$         8330   leo           Tue Apr 16 21:23:48 2013
Plans        8382   192.168.178.200  Tue Apr 16 21:36:41 2013
IPC$         8380   leo           Tue Apr 16 21:36:07 2013
IPC$         8381   leo           Tue Apr 16 21:36:30 2013

No locked files
```

209.1.6 nmblookup

```
nmblookup raspimicha
querying raspimicha on 192.168.178.255
192.168.178.36 raspimicha<00>
```

209.1.7 smbclient

smbclient funktioniert so ähnlich wie ein ftp-Kommandozeilenclient:
Aufruf mit `smbclient servicename`
Optionen: -U user
servicename = //server/service
server = netbios-name service = sharename

> Achtung: Unix- und smb-Passwörter nicht durcheinanderbringen!

```
smbclient //raspimicha/Plans
Enter micha's password:
Domain=[RASPIWORKGROUP] OS=[Unix] Server=[Samba 3.6.6]
smb: \> ls
  .                                   D        0  Tue Apr 16 23:37:52 2013
  ..                                  D        0  Tue Apr 16 23:22:09 2013
  japadapaduuu                        D        0  Tue Apr 16 23:37:52 2013

                59371 blocks of size 131072. 39867 blocks available
smb: \>
```

Samba-Server testen:

```
smbclient -L localhost -U%
```

209.1.8 smbmount

Im Paket smbfs enthalten.

```
smbmount //192.168.178.36/public /mnt -o username=micha,password=mad
```

Hat mit dem Netbiosnamen nicht funktioniert, bei smbclient klappt das dagegen sehr gut.

209.1.9 Das net-Kommando

net rpc info Enter micha's password: Domain Name: WORKGROUP Domain SID: S-1-5-21-2530009695-2628360691-4272010541 Sequence number: 1366292952 Num users: 2 Num domain groups: 0 Num local groups: 0

209.2 NFS Server Configuration

Weight: 4 Description: Candidates should be able to export filesystems using NFS. This objective includes access restrictions, mounting an NFS filesystem on a client and securing NFS.

Key Knowledge Areas

NFS configuration files NFS tools and utilities Access restrictions to certain hosts and/or subnets Mount options on server and client tcpwrappers

Terms and Utilities

```
/etc/exports
exportfs
showmount
nfsstat
/proc/mounts
/etc/fstab
rpcinfo
mountd
portmapper
```

209.2.1 /etc/exports

```
# /etc/exports: the access control list for filesystems which may be exported
#               to NFS clients.  See exports(5).
#
# Example for NFSv2 and NFSv3:
# /srv/homes       hostname1(rw,sync,no_subtree_check) hostname2(ro,sync,no_subtree_check)
#

/home/public 192.168.178.0/24(rw,all_squash)
/home/micha 192.168.178.200/32(rw,no_root_squash,subtree_check)
/home/fritz *.fritz.box(rw,no_root_squash)
```

Wildcards: '*' oder '?' :

***** anonymous = alle; nur ein einzelner '*'!

***.fritz.box** hier steht der '*' für beliebige Zeichen (auch Punkte). Damit werden also alle Host in der Domänen *fritz.box* erlaubt.

Natürlich funzt das mit der fritzBox nicht, weil das dämliche Teil keine DNS-Rückwärtsauflösung beherrscht.

209.2.2 exportfs und nfs-kernel-server

```
exportfs -v                                          #alles auflisten
exportfs -av                                         #alles exportieren, verbose
exportfs -v -u 192.168.178.200/32:/home/micha        #export aufheben
exportfs -v 192.168.178.200/32:/home/micha           #eine resource exportieren
exportfs -vau                                        #alle exports aufheben

/etc/init.d/rpcbind start                            #nur auf raspi manuell starten
/etc/init.d/nfs-kernel-server start                  #auf raspi keine ipv6-unterstuetzung
```

209.2.3 Portmapper und mountd

Funktion:

- Server stellt Sammlungen von Funktionen zur Verfügung
- Jede Sammlung hat einen Namen und eine ID-Nummer:

```
portmapper      100000  portmap sunrpc
rstatd          100001  rstat rstat_svc rup perfmeter
rusersd         100002  rusers
nfs             100003  nfsprog
ypserv          100004  ypprog
mountd          100005  mount showmount
ypbind          100007
walld           100008  rwall shutdown
yppasswdd       100009  yppasswd
```

- Server bildet die Sammlungs-ID auf UDP-Ports ab. Auflisten mit rpcinfo:

```
root@raspiMicha:~# rpcinfo -p localhost
   program vers proto   port  service
    100000    4   tcp    111  portmapper
    100000    3   tcp    111  portmapper
    100000    2   tcp    111  portmapper
    100000    4   udp    111  portmapper
    100000    3   udp    111  portmapper
    100000    2   udp    111  portmapper
    100003    2   tcp   2049  nfs
    100003    3   tcp   2049  nfs
    100003    4   tcp   2049  nfs
    100227    2   tcp   2049
    100227    3   tcp   2049
    100003    2   udp   2049  nfs
    100003    3   udp   2049  nfs
    100003    4   udp   2049  nfs
    100227    2   udp   2049
    100227    3   udp   2049
```

- Möchte eine Anwendung einen RPC machen, fragt sie zunächst beim *portmapper* an, auf welchem UDP-Port die Funktion zur Verfügung steht. Von den Funktionen stehen oft mehrere Versionen bereit.
- der Portmapper-Dienst verwaltet nun genau diese Zuordnung zwischen ID und Port. Kommandos:

```
/sbin/portmap
/sbin/rpcbind #auf Ubuntu/Debian/Raspian
```

- Nachdem klar ist, welcher Port zuständig ist, schickt der Client eine Mount-Anfrage (oder umount request) an den zuständigen Port. Diese Anfrage wird von rpc.mountd verarbeitet:
  - mountd prüft Name/IP des Anfragers gegen die /etc/exports
  - mountd schickt dem Anfrager ein File-Handle für das Stammverzeichnis des exports zurück.

209.2.4 Weitere wichtige Kommandos

showmount Optionen: -a: alle Mounts zeigen, -d: alle gemounteten Verzeichnisse zeigen, -e: Exportliste zeigen

```
showmount -e
Export list for raspiMicha:
/home/world  *
/home/fritz  *.fritz.box
/home/micha  192.168.178.200/32
/home/public 192.168.178.0/24
```

nfsstat gibt Statistik des nfs-Servers aus:

```
root@raspiMicha:~# nfsstat -4l
nfs v4 server        total:      135
------------ ------------- --------
nfs v4 server         null:        3
nfs v4 server     compound:      132

nfs v4 servop        total:      318
------------ ------------- --------
nfs v4 servop       access:       15
nfs v4 servop       create:        2
nfs v4 servop      getattr:      118
nfs v4 servop        getfh:       19
nfs v4 servop       lookup:       24
nfs v4 servop        putfh:      129
nfs v4 servop    putrootfh:        3
nfs v4 servop      readdir:        3
nfs v4 servop     readlink:        2
nfs v4 servop    restorefh:        1
nfs v4 servop       savefh:        2
```

rpcinfo `rpcinfo -p localhost` gibt die Portmap aus (s.o.)

209.2.5 NFS-Client

```
mount -o rsize=8192,wsize=8192,hard,intr raspimicha.fritz.box:/home/micha /mnt/homeMicha
```

210 Network Client Management

210.1 210.1 DHCP configuration

Weight: 2 Description: Candidates should be able to configure a DHCP server. This objective includes setting default and per client options, adding static hosts and BOOTP hosts. Also included is configuring a DHCP relay agent and maintaining the DHCP server.

Key Knowledge Areas
DHCP configuration files, terms and utilities Subnet and dynamically-allocated range setup
Terms and Utilities

```
    dhcpd.conf
    dhcpd.leases
    /var/log/daemon.log
    /var/log/messages
    arp
    dhcpd
```

210.1.1 ISC-dhcp-server

```
default-lease-time 21600;
max-lease-time 43200;
ddns-update-style none;
option subnet-mask 255.255.255.0;
option broadcast-address 192.168.178.255;
option routers 192.168.178.36;
option domain-name-servers 192.168.178.36;
option domain-name "ras.pi.";
option ntp-servers 192.168.178.36;
#
subnet 192.168.178.0 netmask 255.255.255.0 {
       range 192.168.178.10 192.168.178.20;
       range 192.168.178.100 192.168.178.200;
}
#
host lenovoX121 {
     hardware ethernet 04:7d:7b:4d:bd:18;
     fixed-address 192.168.178.99;
     option host-name "piclient";
}
```

210.1.2 dhclient

```
dhclient -v eth0    #lease suchen
dhclietn -r         #release

root@michel:~# dhclient -r && dhclient -v eth0
Internet Systems Consortium DHCP Client 4.1-ESV-R4
Copyright 2004-2011 Internet Systems Consortium.
All rights reserved.
For info, please visit https://www.isc.org/software/dhcp/

Listening on LPF/eth0/04:7d:7b:4d:bd:18
Sending on   LPF/eth0/04:7d:7b:4d:bd:18
Sending on   Socket/fallback
DHCPDISCOVER on eth0 to 255.255.255.255 port 67 interval 3
DHCPREQUEST of 192.168.178.23 on eth0 to 255.255.255.255 port 67
DHCPOFFER of 192.168.178.23 from 192.168.178.1
DHCPACK of 192.168.178.23 from 192.168.178.1
bound to 192.168.178.23 -- renewal in 419801 seconds.
```

210.1.3 dhcpd.leases

Speicherort: `/var/lib/dhcp/dhcpd.leases`

```
# The format of this file is documented in the dhcpd.leases(5) manual page.
# This lease file was written by isc-dhcp-4.2.2

lease 192.168.178.182 {
  starts 0 2013/04/21 18:07:20;
  ends 6 2013/07/20 18:07:20;
  cltt 0 2013/04/21 18:07:20;
  binding state active;
  next binding state free;
  rewind binding state free;
  hardware ethernet 60:fb:42:4b:94:48;
  uid "\001`\373BK\224H";
}
server-duid "\000\001\000\001\031\005\203u\270'\353\244\3752";
```

210.2 210.2 PAM authentication

Weight: 3 Description: The candidate should be able to configure PAM to support authentication using various available methods.

Key Knowledge Areas

PAM configuration files, terms and utilities passwd and shadow passwords

Terms and Utilities

```
    /etc/pam.d
    pam.conf
    nsswitch.conf
    pam_unix
    pam_cracklib
    pam_limits
    pam_listfile
```

210.2.1 PAM

- PAM ist eine API
- Programme, die eine Authentifizierung benötigen, werden gegen die PAM-Lib gelinkt
- die eingebundenen Routinen erledigen die Authentifizierung
- für jedes PAM-gelinkte-Programm, gibt es in `/etc/pam.d/` eine Konfigurationsdatei

/etc/pam.conf und /etc/pam.d/

Ist das Verzeichns /etc/pam.d vorhanden, wird eine evtl. vorhandene Datei /etc/pam.conf ignoriert.

Syntax von `/etc/pam.conf` und den Dateien in `/etc/pam.d`

- `pam.conf` enhält eine Liste von Regeln
- jede Regel steht in einer eigenen Zeile

- jede Zeile hat das Format:
  `service type control module-path module-arguments`

  In den pam.d-Dateien entfällt die erste Spalte `service`. In diesem Fall ist der *Dateiname* der Servicename bzw. Name eines Kommandos. Z.B. die Datei `login` für den login-Service (s.u.) oder Datei `su` für das Kommando su

- **Ganz wichtig:** eine Anzahl von Regeln kann zusammengefasst werden, um die Ergebnisse einer Reihe von PAMs für eine gegebene Authentifizierungsaufgabe zu zusammenzufassen.

Bedeutung der Spalten:

1. . Spalte: Modul**typ**

   auth Dieses Modul prüft die Identität eines Benuzter mit einer Passwortabfrage und kann zusätzlich Gruppenzugehörigkeit vergeben.

   account Dieses Modul prüft Bedingungen, sich einloggen zu dürfen ein. Z.B. kein grafischer Login für *root*, login nur zu bestimmten Uhrzeiten usw.

   session Dieses Modul setzt die Arbeitsumgebung, mounts, loggin usw. für den User

   password Modul für die Verwaltung der Passwörter

2. . Spalte: Der Wert in der zweiten Spalte regelt das Verhalten der PAM-API, wenn die Authentifizierungsaufgaben des Moduls fehlschlagen:

   required Fehlschlag des PAM führt zu einem Abbruch der Authentifizierungsverfahrens.

   requisite Ebenso Abbruch des PAM, aber die Kontrolle wird an die Anwendung, die das PAM verwendet, zurückgegeben.

   sufficient Authentifiziert das PAM erfolgreich, wird die Kontrolle wieder an die Anwendung zurückgegeben und die anderen Module werden nicht mehr angewendet.

   optional Die Authentifizierung über dieses PAM ist nur entscheidend, wenn es das einzige Modul in der Modulkette für diesen Service ist.

   Neben diesen einfachen Werten, gibt es noch Listen mit Return-Codes. Beispiel: `[success=1 default=ignore]` Die Zahl '1' bedeutet: 1 nächstes Modul in der Kette überspringen

3. . Spalte: Pfad zum PAM-(M)odul

4. . Spalte: Argumente, die dem Modul mitgegeben werden.

Datei `/etc/pam.d/login`

```
root@raspiMicha:/etc/pam.d# grep ^[^#] /etc/pam.d/login
auth       optional   pam_faildelay.so  delay=3000000
auth [success=ok new_authtok_reqd=ok ignore=ignore user_unknown=bad default=die] pam_securetty.so
auth       requisite  pam_nologin.so
session [success=ok ignore=ignore module_unknown=ignore default=bad] pam_selinux.so close
session       required   pam_env.so readenv=1
session       required   pam_env.so readenv=1 envfile=/etc/default/locale
@include common-auth
auth       optional   pam_group.so
session    required   pam_limits.so
session    optional   pam_lastlog.so
session    optional   pam_motd.so  motd=/run/motd.dynamic
session    optional   pam_motd.so
session    optional   pam_mail.so standard
@include common-account
@include common-session
@include common-password
session [success=ok ignore=ignore module_unknown=ignore default=bad] pam_selinux.so open

root@raspiMicha:/etc/pam.d# grep ^[^#] /etc/pam.d/common-auth
auth        [success=1 default=ignore]        pam_unix.so nullok_secure
auth        requisite                        pam_deny.so
auth        required                        pam_permit.so

root@raspiMicha:/etc/pam.d# grep ^[^#] /etc/pam.d/common-account
account        [success=1 new_authtok_reqd=done default=ignore]        pam_unix.so
account        requisite                        pam_deny.so
account        required                        pam_permit.so

root@raspiMicha:/etc/pam.d# grep ^[^#] /etc/pam.d/common-session
session        [default=1]                        pam_permit.so
session        requisite                        pam_deny.so
session        required                        pam_permit.so
session        required        pam_unix.so
session        optional                        pam_ck_connector.so nox11

root@raspiMicha:/etc/pam.d# grep ^[^#] /etc/pam.d/common-password
password        [success=1 default=ignore]        pam_unix.so obscure sha512
password        requisite                        pam_deny.so
password        required                        pam_permit.so
```

Wichtige Module:

pam_access Festlegung, wer von wo sich einloggen darf, Konfiguration in `/etc/security/access.conf`

pam_cracklib Schwache Passwörter aufspüren, Konfiguration in `/etc/login.defs`

pam_deny Kein Zugang möglich. Es wird immer deny zurück geliefert.

pam_issue Funktionialität von `/etc/issue` bei lokaler Anmeldung nachbilden.

pam_ldap LDAP-Authentifizierung

pam_limits Limits für Benutzer festlegen (z.B. Anzahl der Prozesse, Prozess-Nice Werte, etc.)

pam_mail Nach dem login prüfen: sind neue Mails da?

pam_mkhomedir Homeverzeichnis anlegen, wenn nicht vorhanden

pam_nologin Kein Login möglich, wenn Datei `/etc/nologin` existiert. Ausnahme: root kann sich trotzdem anmelden.

210.3 210.3 LDAP client usage

Weight: 2 Description: Candidates should be able to perform queries and updates to an LDAP server. Also included is importing and adding items, as well as adding and managing users.

Key Knowledge Areas

LDAP utilities for data management and queries Change user passwords Querying the LDAP directory

Terms and Utilities

```
ldapsearch
ldappasswd
ldapadd
ldapdelete
```

211 211: E-Mail Services

211.1 211.1 Using e-mail servers

Weight: 3 Description: Candidates should be able to manage an e-mail server, including the configuration of e-mail aliases, e-mail quotas and virtual e-mail domains. This objective includes configuring internal e-mail relays and monitoring e-mail servers.

Key Knowledge Areas

Configuration files for postfix Basic knowledge of the SMTP protocol, sendmail, and exim

Terms and Utilities

```
postfix
sendmail
/etc/aliases
/etc/mail/*
/etc/postfix/*
/var/spool/mail
/var/log/
sendmail emulation layer commands
```

211.1.1 postfix-Konfiguration

```
# See /usr/share/postfix/main.cf.dist for a commented, more complete version


# Debian specific:  Specifying a file name will cause the first
# line of that file to be used as the name.  The Debian default
# is /etc/mailname.
#myorigin = /etc/mailname

smtpd_banner = $myhostname ESMTP $mail_name (Ubuntu)
biff = no

# appending .domain is the MUA's job.
append_dot_mydomain = no

# Uncomment the next line to generate "delayed mail" warnings
#delay_warning_time = 4h

readme_directory = no

# TLS parameters
smtpd_tls_cert_file=/etc/ssl/certs/ssl-cert-snakeoil.pem
smtpd_tls_key_file=/etc/ssl/private/ssl-cert-snakeoil.key
smtpd_use_tls=yes
smtpd_tls_session_cache_database = btree:${data_directory}/smtpd_scache
smtp_tls_session_cache_database = btree:${data_directory}/smtp_scache

# sasl-parameters

smtpd_sasl_type = dovecot
smtpd_sasl_path = private/auth
smtpd_sasl_auth_enable = yes
broken_sasl_auth_clients = yes
smtpd_sasl_security_options = noanonymous, noplaintext
smtpd_sasl_tls_security_options = noanonymous
smtpd_tls_auth_only = yes

# restrictions
smtpd_recipient_restrictions = permit_sasl_authenticated,\
permit_mynetworks,reject_unauth_destination,permit

# See /usr/share/doc/postfix/TLS_README.gz in the postfix-doc package for
# information on enabling SSL in the smtp client.

myhostname = msv.wara.de
alias_maps = hash:/etc/aliases
alias_database = hash:/etc/aliases
myorigin = /etc/mailname
mydestination = msv.wara.de, localhost.wara.de, localhost
#relayhost = mbox1.belwue.de
#relayhost = 129.143.2.21
relayhost = mail.belwue.de
#relayhost = 129.143.2.15
mynetworks = 127.0.0.0/8 [::ffff:127.0.0.0]/104 [::1]/128
mailbox_command = procmail -a "$EXTENSION"
mailbox_size_limit = 0
recipient_delimiter = +
inet_interfaces = all
```

211.2 211.2 Managing Local E-Mail Delivery

Weight: 2 Description: Candidates should be able to implement client e-mail management software to filter, sort and monitor incoming user e-mail.

Key Knowledge Areas

procmail configuration files, tools and utilities Usage of procmail on both server and client side

Terms and Utilities

```
~/.procmail
/etc/procmailrc
procmail
mbox and Maildir formats
```

211.2.1 procmail

`procmail` kann so eingerichtet werden, dass es

- entweder gestartet wird, wenn eine neue Mail eintrifft. Dafür den .forward - Mechanismus verwenden: in der Datei `~/.forward` muss folgende Zeile enthalten sein (einschliesslich Anführungszeichen!): `"|exec /usr/bin/procmail"`
- procmail kann auch direkt vom MTA aufgerufen werden (vgl. 211.1.1: `mailbox_command = procmail -a "$EXTENSION"`.
  Die Übergabe erfolgt hier über `/var/spool/mail/<user>`

Konfigurationsdateien:

- Datei `.forward`
- Datei `~/.procmailrc`
- Verzeichnis `~/.procmail`. Die Dateien hierin (sog. *recipes*) müssen von `~/.procmailrc` aus aufgerufen werden.
- Dateien im Verzeichnis

Beispiel für .procmailrc

```
PATH=/usr/local/bin:/usr/bin:/bin
       MAILDIR=$HOME/Mail      #you'd better make sure it exists
       DEFAULT=$MAILDIR/mbox   #completely optional
       LOGFILE=$MAILDIR/procmail.log   #recommended

       :0 c
       * ^From.*(mad.org|alfred)
       fromAlfred

       :0
       * ^From.*Amazon\.de.*
       amazon

       :0
       * ^Subject:.*GLK
       /dev/null

       :0
       * ^From.*dienert@msv.wara.de
       ! dienert@wara.de
```

:0 leitet ein neues *Rezept* ein

:0 c die Mail, auf die die Bedingung zutrifft wird nach mbox kopiert und entsprechend der Regel weiterverarbeitet.

:0 H Egrep den Mail**header**. Das ist die Grundeinstellung

:0 B Egrep den Mail**body**

:0 D Mit 'D' unterscheidet Egrep Gross-/Kleinschreibung

:0 Whc: lockfile

*** \^From.*Amazon\.de.*** Der '*' leitet eine Bedingung ein. Nur eine Bedingung pro Zeile ist erlaubt. Als Regexp wird egrep verwendet. Man darf mehrere Bedingungszeilen schreiben, die sind dann aber **UND**-verknüpft.

.* beliebig viele, beliebige Zeichen (egrep).

^ Anfang einer Zeile (egrep).

! dienert@wara.de an die angegebene Adresse *weiterleiten*.

211.3 211.3 Managing Remote E-Mail Delivery

Weight: 2 Description: Candidates should be able to install and configure POP and IMAP daemons.

Key Knowledge Areas

Courier IMAP and Courier POP configuration Dovecot configuration

Terms and Utilities

```
/etc/courier/*
dovecot.conf
```

211.3.1 dovecot.conf

```
 #erstmal alle kommentare wegstrippen
 awk '!/^ *(#|$)/' dovecot.conf

protocols = imap imaps pop3 pop3s
disable_plaintext_auth = no
log_timestamp = "%Y-%m-%d %H:%M:%S "
ssl_cert_file = /etc/ssl/certs/ssl-cert-snakeoil.pem
ssl_key_file = /etc/ssl/private/ssl-cert-snakeoil.key
mail_location = mbox:~/mail:INBOX=/var/mail/%u
mail_privileged_group = mail
protocol imap {
  mail_max_userip_connections = 20
}
protocol pop3 {
  pop3_uidl_format = %08Xu%08Xv
}
protocol managesieve {
}
auth default {
  mechanisms = plain login
  passdb pam {
  }
  userdb passwd {
  }
  user = root
  socket listen {
    client {
      path = /var/spool/postfix/private/auth
      mode = 0660
      user = postfix
      group = postfix
    }
  }
  !include_try /etc/dovecot/auth.d/*.auth
}
dict {
}
plugin {
}
!include_try /etc/dovecot/conf.d/*.conf
```

212 Topic 212: System Security

212.1 212.1 Configuring a router

Weight: 3 Description: Candidates should be able to configure a system to perform network address translation (NAT, IP masquerading) and state its significance in protecting a network. This objective includes configuring port redirection, managing filter rules and averting attacks.

Key Knowledge Areas

iptables configuration files, tools and utilities Tools, commands and utilities to manage routing tables. Private address ranges Port redirection and IP forwarding List and write filtering and rules that accept or block datagrams based on source or destination protocol, port and address Save and reload filtering configurations

Terms and Utilities

```
/proc/sys/net/ipv4
/etc/services
iptables
routed
```

212.1.1 iptables

Aufbau:

- Die Firewall besteht aus **Tabellen**.
- Eine Tabelle enthält mehrere Filter-**Ketten**.
- Eine Kette besteht aus **Regeln**, die Regeln sind also die Kettenglieder. Die Regeln einer Kette werden nacheinander durchlaufen, trifft eine Regel zu, wird die Kette verlassen.
- Eine Regel endet mit der Angabe eines Sprung-**Ziels**. Das Ziel bestimmt, was mit dem Paket gemacht wird: DROP, ACCEPT, DNAT, ... oder ob man zu einer anderen Kette springt.

Tabellen: tables Es gibt standardmässig die drei Tabellen:

filter ist die Standardtabelle. Ist keine Tabelle angegeben (Option `-t`), wird *filter* verwendet.

nat Die Tabelle für NAT wird mit `-t nat` aufgerufen.

mangle Die Tabelle mangle wird hier ausgespart.

Regeln Eine Regel wird mit **-A *chain*** an die Kette *chain* angehängt.

Ziele: targets bestimmen, wie mit dem Paket verfahren wird. Die Ziele werden mit **-j** oder **–jump** aufgerufen. Es gibt (vordefiniert, Liste nicht vollst.) :

DROP

ACCEPT

MASQUERADE gibt es nur in der nat-Tabelle

DNAT gibt es nur in der nat-Tabelle

Ketten: chains es gibt 5 vordefinierte Ketten (in Blocksatz):

PREROUTING erste Kette, da muss der gesamte Verkehr durch (gut für z.B. *port forwarding*)

INPUT Kette für Pakete, die *für* den Router selbst bestimmt sind

FORWARD Kette für Pakete, die geroutet werden

OUTPUT Kette für Pakete, die *vom* Router selbst stammen

POSTROUTING letzte Kette, da muss der gesamte Verkehr durch (für *nat*)

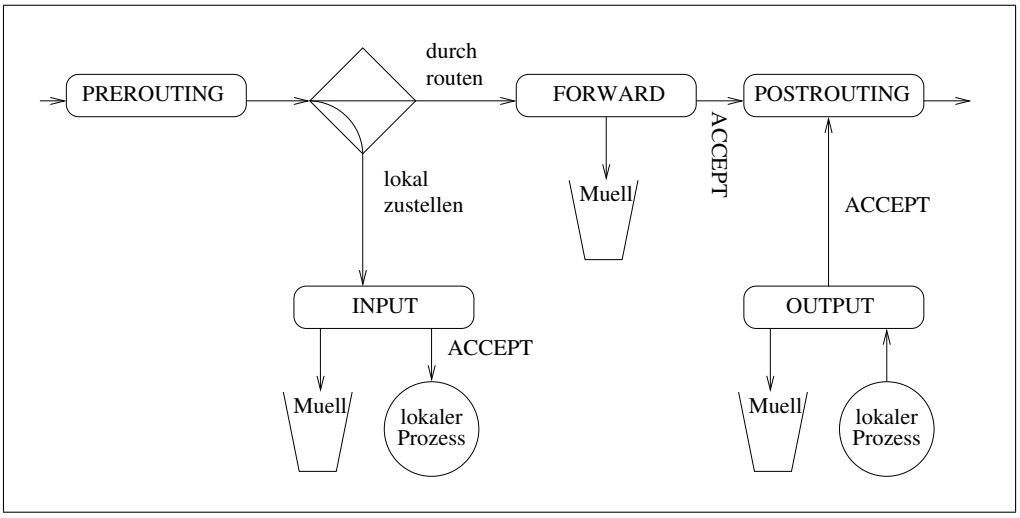

Abbildung 1: Vereinfachtes iptables-Schema

Allgemeine Syntax:

```
iptables iptables [-t table]  -main-option chain [specification] [-j target]
```

main-options:

-A append - Kette hinzufügen

-D drop - Kette löschen

-nL list - alle Ketten ausgeben; -n: keine DNS-Lookups

-F flush - alle Regeln der gewählten Kette löschen

-P policy

-h help

specifications:

-p, –protocol tcp, udp, icmp, all

-s, –source source-ip-address

-d, –destination destination-ip-address

-i, –in-interface Eingangsinterface; nur in den Ketten INPUT, FORWARDING, PREROUTING erlaubt

-o, –out-interface Ausgangsinterface; nur in den Ketten OUTPUT, FORWARDING, POSTROUTING erlaubt

–dport, –destination-port Zielport des Pakets

-m state, –state zustand -m verbindet mit Modulen; hier wird nur das Modul *state* betrachtet. Verbindungszustände sind:

ESTABLISHED Paket gehört zu einer bestehenden Verbindung, egal welche Richtung

INVALID Paket gehört zu keiner bekannten Verbindung

RELATED Paket startet neue Verbindung, die aber zu einer Bestehenden gehört

NEW Paket hat eine neue Verbindung gestartet

targets s.o.
Ausnahme Target DNAT, da kommt noch die Option **–to** hinzu:

```
iptables iptables -t nat  -A PREROUTING [specification] -j DNAT --to ip:port
```

```
#!/bin/sh

IN_IF=eth0
OUT_IF=eth0

IN_NET=10.20.30.0/24
IN_NW=10.20.30.0
IN_BC=10.20.30.255
ANY=0.0.0.0/0

FILESERV=10.20.30.5
WEBSERV=10.20.30.2

iptables -F new_net
iptables -F INPUT
iptables -F FORWARD
iptables -F OUTPUT

#verbindungen von aussen ins innere netz, die erlaubt werden
iptables -N new_net
iptables -A new_net -p tcp --dport imaps --destination $FILESERV --jump ACCEPT
iptables -A new_net -p tcp --dport ssh --destination $FILESERV --jump ACCEPT
iptables -A new_net -p tcp --dport 80 --destination $WEBSERV --jump RETURN

#INPUT traffic geht zum router selbst
#ping, ssh und http erlauben
iptables -A INPUT -m state --state INVALID  --jump DROP
iptables -A INPUT -p icmp --icmp-type echo-request --jump ACCEPT
iptables -A INPUT -m state --state ESTABLISHED,RELATED  --jump ACCEPT
iptables -A INPUT -p tcp --dport ssh --jump ACCEPT
iptables -A INPUT -p tcp --dport http --jump ACCEPT
iptables -A INPUT -i lo --jump ACCEPT
iptables -A INPUT --jump DROP

#alles, was geroutet wird
iptables -A FORWARD -m state --state INVALID --jump DROP
iptables -A FORWARD -m state --state ESTABLISHED,RELATED  --jump ACCEPT
iptables -A FORWARD -i $IN_IF -m state --state NEW --jump ACCEPT
#was von aussen kommt, wird weitergeleitet
iptables -A FORWARD -i $OUT_IF -m state --state NEW --jump new_net
#der rest kommt in die tonne
iptables -A FORWARD --jump DROP

iptables -A OUTPUT -m state --state INVALID --jump DROP
iptables -A OUTPUT -m state --state ESTABLISHED,RELATED --jump ACCEPT
iptables -A OUTPUT -p tcp --jump ACCEPT
iptables -A OUTPUT -p udp --dport domain --jump ACCEPT
iptables -A OUTPUT -p icmp --icmp-type echo-request --jump ACCEPT
iptables -A OUTPUT -o lo --jump ACCEPT
iptables -A OUTPUT --jump DROP

#nat
echo nat einrichten

iptables -t nat -A POSTROUTING -o $OUT_IF --jump MASQUERADE

iptables -t nat -A PREROUTING -i $OUT_IF -p tcp -dport imaps --jump DNAT --to $FILESERV
iptables -t nat -A PREROUTING -i $OUT_IF -p tcp -dport ssh --jump DNAT --to $FILESERV
iptables -t nat -A PREROUTING -i $OUT_IF -p tcp -dport http --jump DNAT --to $WEBSERV

iptables -nL

echo 1 > /proc/sys/net/ipv4/ip_forward

echo routing ist an:
cat /proc/sys/net/ipv4/ip_forward
```

212.1.2 iptables sichern und restaurieren

```
iptables-safe > sicherung
iptables-restore < sicherung
```

212.2 212.2 Securing FTP servers

Weight: 2 Description: Candidates should be able to configure an FTP server for anonymous downloads and uploads. This objective includes precautions to be taken if anonymous uploads are permitted and configuring user access.

Key Knowledge Areas

Configuration files, tools and utilities for Pure-FTPd and vsftpd Awareness of ProFTPd Understanding of passive vs. active FTP connections

Terms and Utilities

```
    vsftpd.conf
    Pure-FTPd command line
```

212.2.1 /etc/vsftpd.conf

```
# standalone (listen=YES) oder inetd:
listen=YES

# lokale benutzer duerfen sich anmelden und haben schreibrechte
local_enable=YES
write_enable=YES

# lokale benutzer sollen nicht aus ihrem home herauskoennen
# funzt nicht, beisst sich mit scheinbar mit write_enable=YES oder so
#chroot_local_user=YES

# anonymer ftp zugang
anonymous_enable=YES

# anonymer user darf PUT ausfuehren; dazu muss AUCH write_enable=YES sein
# UND es muss ein verzeichnis existieren, in dem der ftp-systemuser
# schreibrechte hat, z.b. /srv/ftp:
# drwxr-xr-x 2 root ftp 4096 Apr 29 04:09 ftp
anon_upload_enable=YES
anon_mkdir_write_enable=YES

# anonym hochgeladene dateien einem
# neuen benutzer geben
chown_uploads=YES
chown_username=gast

#dirmessage_enable=YES
#use_localtime=YES
#xferlog_enable=YES
connect_from_port_20=YES
#secure_chroot_dir=/var/run/vsftpd/empty
pam_service_name=vsftpd
rsa_cert_file=/etc/ssl/private/vsftpd.pem
```

212.3 212.3 Secure shell (SSH)

Weight: 2 Description: Candidates should be able to configure and secure an SSH daemon. This objective includes managing keys and configuring SSH for users. Candidates

should also be able to forward an application protocol over SSH and manage the SSH login.

Key Knowledge Areas

OpenSSH configuration files, tools and utilities Login restrictions for the superuser and the normal users Managing and using server and client keys to login with and without password Usage of XWindow and other application protocols through SSH tunnels Configuration of ssh-agent Usage of multiple connections from multiple hosts to guard against loss of connection to remote host following configuration changes

Terms and Utilities

```
ssh
sshd
/etc/ssh/sshd_config
Private and public key files
~/.ssh/authorized_keys
PermitRootLogin
PubKeyAuthentication
AllowUsers
PasswordAuthentication
Protocol
```

212.3.1 sshd_config

```
# What ports, IPs and protocols we listen for
Port 22
# Use these options to restrict which interfaces/protocols sshd will bind to
#ListenAddress ::
#ListenAddress 0.0.0.0
Protocol 2
# HostKeys for protocol version 2
HostKey /etc/ssh/ssh_host_rsa_key
HostKey /etc/ssh/ssh_host_dsa_key
HostKey /etc/ssh/ssh_host_ecdsa_key
#Privilege Separation is turned on for security
UsePrivilegeSeparation yes

# Lifetime and size of ephemeral version 1 server key
KeyRegenerationInterval 3600
ServerKeyBits 768

# Logging
SyslogFacility AUTH
LogLevel INFO

# Authentication:
LoginGraceTime 120
PermitRootLogin yes
StrictModes yes

RSAAuthentication yes
PubkeyAuthentication yes
AuthorizedKeysFile        %h/.ssh/authorized_keys

# Don't read the user's ~/.rhosts and ~/.shosts files
IgnoreRhosts yes
# For this to work you will also need host keys in /etc/ssh_known_hosts
RhostsRSAAuthentication no
# similar for protocol version 2
HostbasedAuthentication no
# Uncomment if you don't trust ~/.ssh/known_hosts for RhostsRSAAuthentication
#IgnoreUserKnownHosts yes

# To enable empty passwords, change to yes (NOT RECOMMENDED)
PermitEmptyPasswords no

# Change to yes to enable challenge-response passwords (beware issues with
# some PAM modules and threads)
ChallengeResponseAuthentication no

# Change to no to disable tunnelled clear text passwords
#PasswordAuthentication yes

X11Forwarding yes
X11DisplayOffset 10
PrintMotd no
PrintLastLog yes
TCPKeepAlive yes
#UseLogin no

#MaxStartups 10:30:60
#Banner /etc/issue.net

# Allow client to pass locale environment variables
AcceptEnv LANG LC_*

Subsystem sftp /usr/lib/openssh/sftp-server

# Set this to 'yes' to enable PAM authentication, account processing,
# and session processing. If this is enabled, PAM authentication will
# be allowed through the ChallengeResponseAuthentication and
# PasswordAuthentication.  Depending on your PAM configuration,
# PAM authentication via ChallengeResponseAuthentication may bypass
# the setting of "PermitRootLogin without-password".
# If you just want the PAM account and session checks to run without
# PAM authentication, then enable this but set PasswordAuthentication
# and ChallengeResponseAuthentication to 'no'.
UsePAM yes
#DenyUsers root
AllowUsers root micha
```

212.3.2 Schlüsselverwaltung

Schlüssel des Servers:

```
root@raspiMicha:/etc/ssh# tree -pu
.
|-- [-rw-r--r-- root    ]  moduli
|-- [-rw-r--r-- root    ]  ssh_config
|-- [-rw-r--r-- root    ]  sshd_config
|-- [-rw-r--r-- root    ]  sshd_config~
|-- [-rw------- root    ]  ssh_host_dsa_key
|-- [-rw-r--r-- root    ]  ssh_host_dsa_key.pub
|-- [-rw------- root    ]  ssh_host_ecdsa_key
|-- [-rw-r--r-- root    ]  ssh_host_ecdsa_key.pub
|-- [-rw------- root    ]  ssh_host_rsa_key
\-- [-rw-r--r-- root    ]  ssh_host_rsa_key.pub
```

Verbindungsaufbau:

- Der öffentliche Schlüssel (`ssh_host_rsa_key.pub`) wird beim Client in `known_hosts` gespeichert.
- Der Server erzeugt eine Zufallszahl (Challenge) und schickt sie dem Client
- Der Client verrechnet den öffentlichen Schlüssel aus `known_hosts` mit der Zufallszahl und schickt das Ergebnis zurück (Response)
- Der Server bestimmt aus der Response wieder den öffentlichen Schlüssel und prüft diesen mit seinem privaten Schlüssel (`ssh_host_rsa_key`)

Schlüssel für `PubkeyAuthentication` erzeugen

```
ssh-keygen -t rsa -b 2048 -f /home/micha/.ssh/ssh_host_key

Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in /home/micha/.ssh/ssh_host_key.
Your public key has been saved in /home/micha/.ssh/ssh_host_key.pub.
The key fingerprint is:
6c:71:e5:50:40:24:1b:17:c5:f9:3a:a7:6f:b2:fc:b1 micha@leo
The key's randomart image is:
 ...
```

Anschliessend:

> Auszug man-page: The user should then copy the public key to `~/.ssh/authorized_keys` in his/her home directory on the remote machine.

Remote Kommando ausführen

```
20:11:52|micha@leo:~$ ssh micha@raspimicha.fritz.box ls
alfred
hierIstHomeMicha
index.html
loop.sh
Mail
nfsRocks
pistore.desktop
public_html
```

Remote X11-Kommando ausführen

```
ssh -X micha@raspimicha.fritz.box xaos
```

212.4 212.4 TCP Wrapper

Weight: 1 Description: Candidates should be able to configure TCP Wrapper to allow connections to specified servers only from certain hosts or subnets.

Key Knowledge Areas

TCP Wrapper configuration files, tools and utilities inetd configuration files, tools and utilities

Terms and Utilities

```
/etc/inetd.conf
/etc/hosts.allow
/etc/hosts.deny
libwrap
tcpd
```

212.5 212.5 Security tasks

Weight: 3 Description: Candidates should be able to receive security alerts from various sources, install, configure and run intrusion detection systems and apply security patches and bugfixes.

Key Knowledge Areas

Tools and utilities to scan and test ports on a server Locations and organisations that report security alerts as Bugtraq, CERT, CIAC or other sources Tools and utilities to implement an intrusion detection system (IDS) Awareness of OpenVAS

Terms and Utilities

```
telnet
nmap
snort
fail2ban
nc
iptables
```

213 Topic 213: Troubleshooting

213.1 213.1 Identifying boot stages and troubleshooting bootloaders

Weight: 4 Description: Candidates should be able to determine the cause of errors in loading and usage of bootloaders. GRUB and LILO are the bootloaders of interest.

Key Knowledge Areas

boot loader start and hand off to kernel kernel loading hardware initialisation and setup daemon/service initialisation and setup Know the different bootloader install locations

on a hard disk or removable device Overwriting standard bootloader options and using bootloader shells

Terms and Utilities

```
    /boot/
    /boot/grub/
    GRUB
    grub-install
    initrd, initramfs
    Master boot record
    /etc/init.d
    lilo
    /etc/lilo.conf
```

213.2 213.2 General troubleshooting

Weight: 5 Description: Candidates should be able to identify and correct common boot and run time issues.

Key Knowledge Areas

/proc filesystem Various system and daemon log files Content of /, /boot , and /lib/modules Screen output during bootup Kernel syslog entries in system logs (if entry is able to be gained) Tools and utilities to analyse information about the used hardware Tools and utilities to trace software and their system and library calls

Terms and Utilities

```
    dmesg
    /sbin/lspci
    /usr/bin/lsdev
    /sbin/lsmod
    /sbin/modprobe
    /sbin/insmod
    /bin/uname
    strace
    strings
    ltrace
    lsof
    lsusb
```

213.2.1 strace

```
root@raspiMicha:/etc/ldap# strace echo "ja" > /tmp/jatrace
execve("/bin/echo", ["echo", "ja"], [/* 16 vars */]) = 0
brk(0)                                  = 0x1091000
uname({sys="Linux", node="raspiMicha", ...}) = 0
access("/etc/ld.so.nohwcap", F_OK)      = -1 ENOENT (No such file or directory)
mmap2(NULL, 8192, PROT_READ|PROT_WRITE, MAP_PRIVATE|MAP_ANONYMOUS, -1, 0) = 0xb6ff3000
access("/etc/ld.so.preload", R_OK)      = 0
open("/etc/ld.so.preload", O_RDONLY)    = 3
   ....
```

strace startet das angegebene Kommando und gibt alle Systemaufrufe und Signale die dieses Kommando abschickt bzw. erhält aus.

213.2.2 ltrace

Schreibt die so-Library-Aufrufe des angegebenen Kommandos mit.

```
root@michel:/etc/ldap# ltrace echo "nein"
__libc_start_main(0x401170, 2, 0x7fff04501c58, 0x403b80, 0x403c10 <unfinished ...>
getenv("POSIXLY_CORRECT")                    = NULL
strrchr("echo", '/')                         = NULL
setlocale(6, "")                             = "de_DE.UTF-8"
bindtextdomain("coreutils", "/usr/share/locale") = "/usr/share/locale"
textdomain("coreutils")                      = "coreutils"
__cxa_atexit(0x401ab0, 0, 0, 0x736c6974756572, 3) = 0
strcmp("nein", "--help")                     = 65
strcmp("nein", "--version")                  = 65
```

213.3 213.3 Troubleshooting system resources

Weight: 5 Description: Candidates should be able to identify, diagnose and repair local system issues when using software from the command line.

Key Knowledge Areas

```
/etc/profile && /etc/profile.d/
/etc/init.d/
/etc/rc.*
/etc/sysctl.conf
/etc/bashrc
/etc/ld.so.conf
or other appropriate global shell configuration files
```

Terms and Utilities

```
/bin/ln
/bin/rm
/sbin/ldconfig
/sbin/sysctl
```

213.4 213.4 Troubleshooting environment configurations

Weight: 5 Description: Candidates should be able to identify common local system and user environment configuration issues and common repair techniques.

Key Knowledge Areas

Core system variables init configuration files init start process cron configuration files Login process User-password storage files Determine user group associations SHELL configuration files of bash Analysing which processes or daemons are running

Terms and Utilities

```
/etc/
/etc/inittab
/etc/rc.local
/etc/rc.boot
/var/spool/cron/crontabs/
/etc/login.defs
/etc/syslog.conf
/etc/passwd
/etc/shadow
/etc/group
/sbin/init
/usr/sbin/cron
/usr/bin/crontab
```