

Diskless-Linux

Betrieb des ATIS-Pools unter Fedora 17 mit über 100 Arbeitsplätzen ohne lokale Festplatten

ATIS - Abteilung Technische Infrastruktur, Fakultät für Informatik



Motivation

- ATIS: zentraler IT-Dienstleister der Fakultät für Informatik

- Studentenpool mit über 100 Arbeitsplätzen, über 2.500 Accounts
 - Dual-Boot unter Windows und Linux
 - Früher
 - Deployment via KickStart
 - Imaging via Clonezilla
 - Langsam, fehleranfällig, erfordert Wartungsfenster
 - Windows Deployment „fummelt“ am MBR und der Partitionstabelle herum
 - Nachinstallationen / Patches via cron-Krücken
 - Installationen sind zu 99,9... % identisch

Ziel

- Booten der Rechner übers Netz
- Betriebssystem via NFS vom Server
- Einfache Wartung
 - Alleinige Pflege **eines** Clientensystems in einer „chroot“-Umgebung auf dem Server
- Vorhandene lokale Festplatte wird nicht angefasst
 - Windows kann sich darauf austoben
- **Kein** kastriertes Linux à la X-Terminal / Thin-Client bei dem fast alles im RAM und der CPU des Servers läuft
 - Der lokale RAM und CPU werden genutzt
- Die Rechner sind diskless und stateless
 - <RESET> stellt den originären Zustand wieder her
- Heimatverzeichnisse (/home) und proprietäre, nicht-distributions-SW (/opt) kommen wie gewohnt via NFS

Voraussetzung: PXE-Boot

■ PXE-Boot

(Vortrag F. Toussaint, (K)IT-Expertenkreis 18.4.2012)

- Netzkarte sendet DHCP-Request
- DHCP-Server antwortet mit IP-Konfiguration
- DHCP-Server verweist auf TFTP-Server und PXE-Konfiguration



- Via PXE-Menü wird Kernel und initrd vom TFTP-Server geladen
- Kernel bootet
- initrd mountet „/“ vom NFS-Server
- „GO“

Schwierigkeiten / Probleme

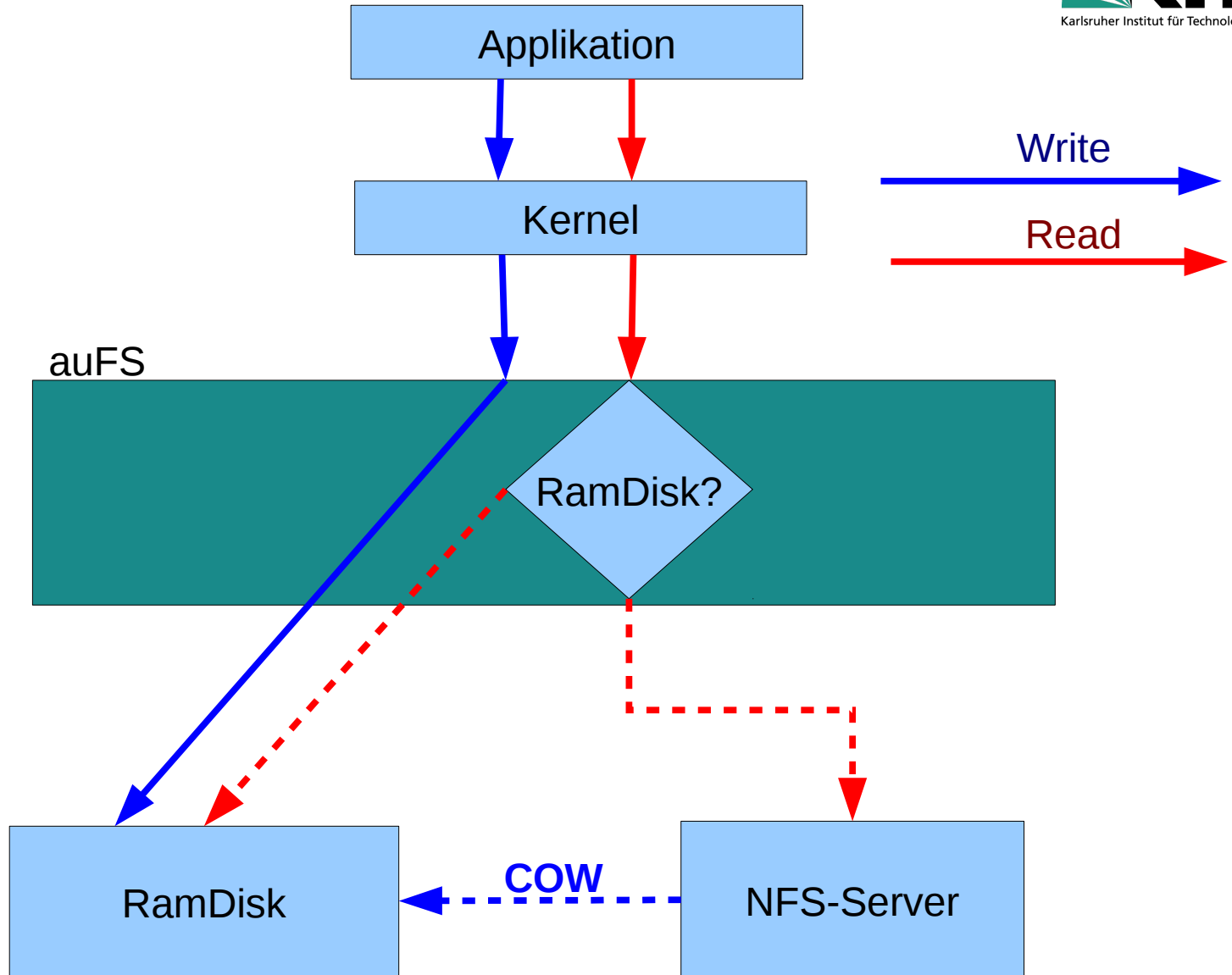
- 1 Image via rw-NFS für alle Clients ?
 - /var und auch /etc (!) werden beschrieben
- Je Client ein Image ?
 - Skaliert nicht
- Unterschiedliche Hardware-Konfigurationen
 - 3 Generationen von BaWü-PCs
 - Unterschiedliche Grafik-Karten (ATI und Nvidia)
 - Exoten-Lösungen
 - 3 Monitore
 - US-Tastatur
- Unterschiedliche Software-Konfigurationen
 - Netze mit unterschiedlichen Authentifizierungen
 - Unterschiedliche LDAP-Server / -Zweige
 - NIS
 - Bibliotheksrechner im Kiosk-Mode

Einschub: unionFS / auFS

- *unionFS* bzw. *auFS* sind Copy-On-Write-Filesystemtreiber, die sich zwischen Kernel und das eigentlichen Filesystem (hier NFS) schieben
- Schreibzugriffe werden abgefangen und in eine RamDisk umgeleitet
- Lesezugriffe schauen zuerst in der RamDisk nach und danach auf dem NFS-Server
- Damit ist die ReadOnly-NFS-Freigabe aus Sicht des Clienten beschreibbar
- <RESET> vergisst alles wieder

- Findet auch Verwendung in den diversen Linux-Live-CDs
- unionFS
 - Alt und buggy, läuft im Userland via fuse
- auFS (**A**nother **U**nion **F**ile **S**ystem)
 - Neuimplementierung als Kernel-Modul
 - z.B. Knoppix seit 5.1

auFS



Lösung mit auFS

- Mit auFS bekommen wir das Problem mit `/var/log` gelöst:
 - Logs landen im RAM
 - sind beim Reboot „weg“
 - zusätzliches Logging zu einem Remote-syslog-Server
- Hängt man statt `/var/log` gleich „/“ über auFS ein, kann man
 - sogar lokal Pakete für Tests nachinstallieren
 - ...bis zum Reboot
 - und hat man DHCP-Schreibzugriffe auf `/etc` auch gelöst
- **Aber wie differenziert man `/etc` ?**
- Idee:
 - Man nimmt das „normale“ `/etc` per NFS und kopiert dann aus einem extra NFS-Verzeichnis, in dem nur die Deltas liegen, diese über das Default-`/etc` drüber,
 - Funktioniert wunderbar wg. auFS

Konfigurationsgruppen

■ NFS-Freigabe auf dem Server

- /diskless/root/etc
 - /usr
 - /var
 - ...

■ Aufbau einer Config-Gruppe „ohopp“, hier liegen nur die Deltas

- /diskless/conf/**ohopp**/etc
 - ...

■ Beispiel einer Gruppe:

```
# find ./diskless/conf/ohopp/  
./diskless/conf/ohopp/  
./diskless/conf/ohopp/etc  
./diskless/conf/ohopp/etc/ssh  
./diskless/conf/ohopp/etc/ssh/sshd_config  
./diskless/conf/ohopp/etc/sysconfig  
./diskless/conf/ohopp/etc/sysconfig/i18n  
./diskless/conf/ohopp/etc/sysconfig/keyboard  
./diskless/conf/ohopp/etc/sudoers.NOT  
./diskless/conf/ohopp/etc/rc.d  
./diskless/conf/ohopp/etc/rc.d/rc.local  
./diskless/conf/ohopp/etc/X11  
./diskless/conf/ohopp/etc/X11/xorg.conf
```

Wie kommen die Deltas zum Client ?

- Der DHCP-Server kann sog. Option-Strings an den Client übergeben
- Diese werden von der initrd abgefragt und dann die entsprechenden config-Delta-Dateien über den „lokalen“ (NFS-) Dateibaum mit rsync kopiert.
- Diese landen somit via auFS im RAM des Klienten
- Damit kann man auch Konfigurationen „stapeln“
 - d.h. mehrere config-Gruppen pro Client möglich

DNS / DHCP Verwaltungstool der Informatik

- DNSadm entspricht dem DNSVS am Campus Süd
- Dient auch zur Pflege des DHCP-Servers

Modify host 141.3.12.50 - Mozilla Firefox

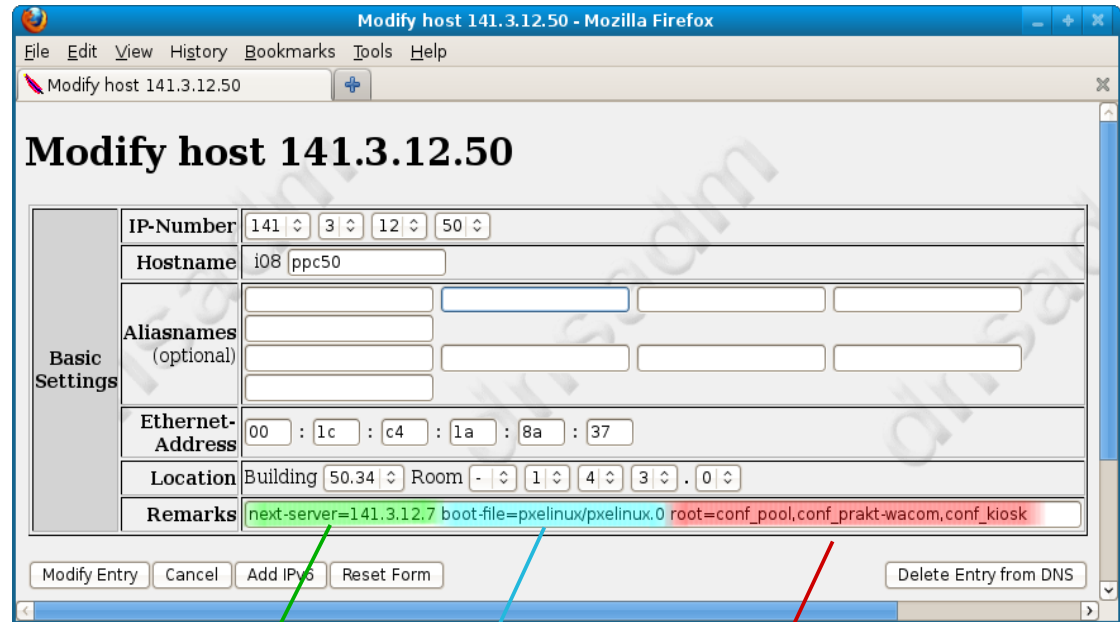
File Edit View History Bookmarks Tools Help

Modify host 141.3.12.50

Modify host 141.3.12.50

Basic Settings	IP-Number	141	3	12	50					
	Hostname	i08 ppc50								
	Aliasnames (optional)	<input type="text"/> <input type="text"/> <input type="text"/> <input type="text"/>								
	Ethernet-Address	00	1c	c4	1a	8a	37			
	Location	Building	50.34	Room	-	1	4	3	.	0
	Remarks	next-server=141.3.12.7 boot-file=pxelinux/pxelinux.0 root=conf_pool,conf_prakt-wacom,conf_kiosk								

DNSadm und conf-Gruppen



■ dhcpd.conf:

```

host i08ppc50 {
    hardware ethernet 00:1c:c4:1a:8a:37;
    fixed-address i08ppc50.atis-stud.uni-karlsruhe.de;
    option host-name "i08ppc50";
    option dhcp-client-identifier "i08ppc50";
    option netbios-node-type 8;
    next-server 141.3.12.7;
    option root-path "conf_pool,conf_prakt-wacom,conf_kiosk";
    filename "pxelinux/pxelinux.0";
}
  
```

Client-Boot

```

i08diskless-PoolPC on iraesx3.ira.uni-karlsruhe.de
File View VM
[ 11.517941] pcnet32: pcnet32.c:v1.35 21.Apr.2008 tsbogend@alpha.franken.de
[ 11.518469] pcnet32: PCnet/PCI II 79C970A at 0x2000, 00:50:56:89:2f:16 assign
ed IRQ 18
[ 11.519191] pcnet32: eth0: registered as PCnet/PCI II 79C970A
[ 11.519713] pcnet32: 1 cards_found
RTNETLINK answers: File exists
Starting dhcp for interface eth0
[ 12.387179] pcnet32 0000:02:00.0: eth0: link up
[ 19.507660] dracut: Mounted root filesystem 141.3.12.7:/diskless/F17/root/
11 Oct 07:09:04 ntpdate[378]: adjust time server 141.3.10.90 offset -0.034765 se
c
*** || Applying configuration target conf_maus*** || Applying configuration targ
et conf_vmware*** || Applying configuration target conf_i08[ 25.633788] aufs a
u_opts_parse:1126:mount[395]: ignored debug=1
[ 25.633984] aufs test_add:261:mount[395]: uid/gid/perm /sysroot 0/0/0555, 0/0
/0755
rpcbind: rpcbind terminating on signal. Restart with "rpcbind -w"
[ 25.803767] dracut: Switching root
[ 26.259434] SELinux: Disabled at runtime.
[ 26.259634] type=1404 audit(1349939345.415:2): selinux=0 audit=4294967295 ses=
4294967295


Welcome to Fedora 17 (Beefy Miracle)!
  
```

Zusammenfassung (I)

- Produktiver Betrieb von über 100 PCs im ATIS-Pool
 - ~50% booten Linux per Default
- NFS-Server: „Standard“ 2 CPUs, 4 Kerne, 8 GB RAM
 - NFS Parameter hochschrauben
 - Nur Last, wenn „viele“ Clients gleichzeitig booten („iowait“)
- Clients mind. 2 GB, besser 4 GB RAM
- Clients überleben den Reboot des Servers !
- Clients verhalten sich erratisch bei Updates von systemnaher Software
 - z.B. glibc , aber auch Firefox
 - Pool-PCs werden jede Nacht neu gebootet
 - „entleert“ auch die RamDisk

Zusammenfassung (II)

- Pflege in einer chroot-Umgebung auf dem Server
 - „yum install XYZ“ / „yum update“
 - Sofort auf den Clients verfügbar

- Rollout einer neuen Fedora-Release: boot und gut 

- Über die Konfigurationsgruppen werden abgebildet
 - Pool
 - Arbeitsplatz-PCs der ATIS-Mitarbeiter
 - Informatik-Bibliothek im Kiosk-Mode
 - eine Forschungseinrichtung
 - individuelle Hardware-Lösungen

- Aktuelle Baustelle
 - NFS4 + Kerberos