



Einer für alle Linux-Printserver für (fast) treiberfreie Windows-Clients

Die nächsten 40 Minuten:

- Die Vorgeschichte
- Die einfache Variante (CUPS)
- Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)
- Zusammenfassung

Einer für alle – Linux–Printserver für (fast) treiberfreie Windows –Clients



Die Vorgeschichte



Worüber wir sprechen:

- Umgebungen mit >100 Druckern und > 10 Typen
- Nutzer installieren und administrieren nichts
- Aufträge laufen zwingend über einen Printserver (nicht direkt auf Boxen, JetDirect etc.)
- lokal freigegebene Drucker müssen unterstützt werden (Windows–Shares)

So druckt Windows:

- Für jeden anzusprechenden Druckertyp ein eigener Treiber **lokal** erforderlich
- Jeder Drucker(eintrag) wird mit einer Queue auf dem Printserver verbunden (SMB direkt, SMB über *LPT+net use*, IPP, TCP etc.)
- PC wandelt den Auftrag in das richtige Zielformat um, **Printserver** hat weitgehend **Spoolfunktion**

Das bereitet Probleme:

- Lokale Installationen/Einstellungen sind unter Windows sehr schlecht zentral zu pflegen
- Mobile Mitarbeiter müssten praktisch alle Treiber vorhalten, da sie die jeweils vor Ort angetroffenen Drucker nutzen
- Netzwerknachricht ist (wenn sie denn überhaupt am richtigen PC ankommt) wenig aussagekräftig
- Windows–Printserver ist nur grafisch kontrollierbar

Das spezielle Problem der Druckerzuordnung:

- Direkte Zuordnung eines Netzdruckers scheidet aus, da überhaupt nicht mehr zentral veränderbar
- Ausweg: Druckereinträge an lokale LPTs binden, Netzdrucker (z.B. über Login–Skript) den LPTs zuordnen
- Funktioniert nicht für die Clients, die sich mit Druckerfreigaben von noch nicht eingeschalteten PCs verbinden wollen

Hier wollen wir hin:

- Ausstatten der Clients mit einem einheitlichen, universellen Druckertreiber
- Administration der Netzdrucker und –zuordnung so weit wie möglich zentralisieren und vereinfachen
- Netzwerknachrichten bereitstellen, die die Fehlersuche unterstützen („... gedruckt hat er eigentlich, aber es kommt nichts raus ...“ –doch es kommt was raus, aber nicht in Chemnitz)

Das könnte eine Lösung sein:

- Einsatz von modellunabhängigen Postscript–Treibern + Umwandlung in Zielformat am Druckserver = Nachbildung des uralten Druckprinzips der Unix–Welt
- Und for free: PS–Features, die mancher „Originaltreiber“ nicht geboten hätte (n Seiten pro Blatt, Wasserzeichen, ps2pdf für PDF statt Papier etc.)

Vorarbeiten am Client:

- Postscripttreiber am PC installieren
- Positiv getestet:
 - Generic Postscript Treiber aus „Adobe Universal Postscript Windows Driver Installer“ von www.adobe.com (kann auch CUPS–PPDs benutzen)
 - HP LaserJet 4000 Series PS / 4050 Series PS
 - Postscript benötigt am Server etwas „Nacharbeit“

Einer für alle – Linux–Printserver für (fast) treiberfreie Windows –Clients

Die einfache Variante (CUPS)



Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients
Die einfache Variante (CUPS)

Printserver installieren:

- Entsprechende PPDs von www.linuxprinting.org installieren (Ghostscript–Versionen)

- Samba informieren (`smb.conf`)

```
[global]
printing = cups
[printers]
print command = (grep -v "^PJL" %s | \
grep -v "%-12345X" | lpr -P%p; rm -f %s)
```

- **Achtung!** (`cupsd.conf`)

```
Printcap /etc/printcap
```

Einer für alle – Linux–Printserver für (fast) treiberfreie Windows –Clients

Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)



Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients
Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)

Ganz kurz erklärt:

- `/etc/printconf` –Beschreibung der Printserverqueues
- Prinzipieller Aufbau:

```
queuename | alias_1 | ... | alias_n  
:option_1  
:...:  
:option_n
```
- 1. Alias erscheint als Beschreibung in `net view`
- Zahlreiche Optionen und Parameter, um eigene erweiterbar

Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients
Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)

Ganz kurz erklärt II:

- Die wichtigsten Optionen für den Einsteiger:
:sd= Spoolverzeichnis der Queue
:filter= Filter zur Zielformatierung
:filter_options= Optionen für :filter
:lp= der lokale Druckeranschluß
:rm= ein anderer Printserver
:rp= die Queue auf dem anderen Server
:meine_option= eigene Option, \${meine_option} in
:filter_options :filter_options
- Änderungen mit *checkpc -f* prüfen und mit *lpc reread* aktivieren

Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients
Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)

Filter:

- Filter wandeln das vom Client gelieferte Postscript in das Zielformat des Druckers um (PCL, ESC etc.)
- Gute Ergebnisse mit `lpdomatic + foomatic-gswrapper` von www.linuxprinting.org, dort findet man auch die nötigen Descriptionfiles für die Drucker
- Eigentlichen Aufruf in Skript verpacken, das erst den Typ des Spoolfiles prüft → „Originaltreiber“ weiter verwendbar

Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients
Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)

Drucken auf Windows–Freigaben:

- Windows–Benutzer anlegen, der auf die Freigabe drucken darf

- Verbindungs– und Anmeldeinformationen

```
( /spoolverzeichnis / .config )
```

```
server=<NetBIOS-Name des Windows-PC>
```

```
service=<Freigabename>
```

```
winuser=<Benutzername für Anmeldung an der Freigabe>
```

```
password=<Paßwort von $winuser>
```

- In *smbprint* ändern:

```
smbclient ... -U $server ZU smbclient ... -U $winuser
```

- Druckeranschluß in /etc/printcap

```
:lp= | /pfad/zu/smbprint
```

Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients
Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)

Priorisierung von Druckaufträgen:

- `lpr -C[A..Z]*`
- Der Großbuchstabe hinter `-C` entscheidet über die Priorität des Druckauftrages (`A`–niedrigste, `Z`–höchste)
- Möglicher Anwendungsfall: kleine Aufträge vor großen einsortieren

- **Simple Beispiel:**

```
lpr -C`echo <jobgroesse> | \
```

```
awk '{printf "%c", 90-length($1)}' `...`
```

Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients
Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)

Lastverteilung:

- Load–Balancing–Queues geben ihre Druckaufträge an Load–Balancing–Server weiter
- Möglicher Anwendungsfall: Druckaufträge gleichmäßig über mehrere Festplatten/Mountpoints verteilen
- LB–Queue und –Server (/etc/printcap)

```
myqueue: . . . : sv=myserver_lb1 , myserver_lb2  
myserver_lb1 : . . . : ss=myprinter  
myserver_lb2 : . . . : ss=myprinter
```

Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients
Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)

Lastverteilung II:

- LB–Server werden zyklisch abgefragt, der erste untätige erhält den Auftrag
- Auswahlverfahren kann durch ein :chooser–Programm beeinflusst werden (/etc /printcap)

```
myqueue: . . .  
:sv=myserver_lb1,myserver_lb2  
:chooser=/pfad/zu/mychooser
```

**Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients
Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)**

Lastverteilung :chooser :

- **Parameter:** alle Filter–Parameter
- **Stdin:** kommagetrennte Liste der LB–Server in der Reihenfolge, wie sie der Load–Balancer abgefragt hätte
- **Stdout:** Name des zu verwendenden LB–Servers

Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients
Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)

Routing:

- Ähnlich dem Load–Balancing, allerdings findet Routing bereits beim **Einreihen** in die LB–Queue statt → Joboptionen können noch verändert werden
- Möglicher Anwendungsfall: Druckaufträge auf unterschiedliche Medien spoolen (Ramdisk, Festplatte)
- Parameter: alle Filter–Parameter
- Stdin: Inhalt des Job–Controlfiles (Joboptionen)

Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients
Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)

Routing II:

- **Stdout: Routinganweisungen**

```
dest <lb-server> # Zielqueue  
copies n # Kopienanzahl auf dieser Queue  
priority [A..Z] # (neue) Priorität  
C<zeile> <wert> # setze Zeile <zeile> des Controlfiles  
# auf den Wert <wert>  
# Ende des Routingblocks  
end
```
- Für einen Druckjob können mehrere Routen mit unterschiedlichen Anweisungen definiert werden, jede wird mit *end* abgeschlossen

Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients
Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)

- „globale“ printcap–Einstellungen (includes):
- Optionen, die von mehreren Queues mit gleichen Werten verwendet werden, können zu „includes“ zusammengefaßt werden
- Includes sind aufgebaut wie Queuebeschreibungen, „Queuename“ beginnt aber mit einem Punkt
- Einbinden erfolgt durch Option :tc=

```
.common:sd=/var/spool/lpd/%P:force_localhost  
.hp:filter=/usr/libexec/filters/hp_filter  
myqueue|Eine Demoqueue:tc=.common,.hp:lp=/dev/null
```
- Nach dem include angegebene Optionen haben Vorrang vor importierten (Overriding)

Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients
Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)

Dynamische printcap:

Printcap–Einträge können auch von Programmen

bereitgestellt werden (/etc/lpd.conf)

...
printcap_path=|/pfad/zum/programm

- Parameter: Filter–Parameter
- Stdin: Name der angeforderten Queue
- Stdout: Queuebeschreibung im printcap–Format
- Achtung! Die printcap wird bei allen Spoolvorgängen und bei Lpc–Aufrufen gelesen, das Programm sollte also schnell antworten

Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients
Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)

Druckbenachrichtigung:

- Reaktion auf Fehler möglich durch Option
`send_failure_action (/etc/lpd.conf)`
...
`send_failure_action=/pfad/zum/programm`
...
- Positives Ergebnis wird nicht so komfortabel zurückgemeldet (evtl. Returncode von lpr benutzen)
- Netzwerknachricht an Windows–Clients wird von `smbClient` unterstützt
`echo ${MEINE_NACHRICHT} | smbclient -M ${NetBIOSClient}`

Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients
Mehr Arbeit, mehr Optionen, mehr Freude (LPRng)

Druckbenachrichtigung II:

- NetBIOS–Informationen müssen schon in der smb.conf abgegriffen werden

```
/etc/samba/smb.conf
[printers]
    print command = /pfad/zu/mylpr %p %s %U %m %h %T

/pfad/zu/mylpr
NetBIOSUser=$3;
NetBIOSClient=$4;
PrintserverNBName=$5;
Druckzeit=$6;

lpr -P$1 -r $2
```

Einer für alle – Linux–Printserver für (fast) treiberfreie Windows –Clients

Zusammenfassung



- Unix–/Linux–Printserver kann signifikante Druckprobleme in Windows–Umgebungen vermeiden und interessante Features bereitstellen
- Sowohl CUPS als auch LPRng sind als Printserver geeignet, wobei CUPS deutlich leichter und übersichtlicher zu konfigurieren, LPRng im Gegenzug jedoch wesentlich flexibler ist.

Verwendete Releases

- Red Hat Linux 7.1 Kernel 2.4.9–12
- CUPS 1.1.12–0
- LPRng–3.8.6
- Ghostscript–5.50–17
- Samba 2.2.2

Einer für alle –Linux–Printserver für (fast) treiberfreie Windows –Clients

Herzlichen Dank den Zuhörern und allen Open–Source–Entwicklern und noch viel Spaß auf dem 4. Chemnitzer Linux–Tag.

