

How to manually setup printing using lpd, ... on MiNT
revision 0.0.1 - Heinz Schmidt - initial version

I guess the most clean way to setup lpd is to make / make install it from the sources.

How ever this is a self made documentation how to put it all together manually. It is based on the lpd 5.90 sources and binaries which come with MiNT 98.

In the late 90is there was the commercial distribution "MiNT 98" which in cluded a lot of network programs and services. Some of the binaries are takten from this distribution.

For this documentation an existing EasyMint 1.72 installation was updated with MiNT kernel 1-17-0. The filesystem structure between MiNT 98 and EasyMint are a bit different so there are some directories which have to be created on EsayMint.

It's assumed that Mint including MintNet is setup and working properly

Files:

```
/usr/etc/pac          Owner: root      Groups: wheel, pac      permissions: 755      man page: y
printer accounting information
/var/account/?acct    raw accounting files
/var/account/?_sum    summary accounting files
/etc/printcap         printer description file
```

```
/usr/etc/lptest      Owner: root      Groups: wheel, lptest   permissions: 755      man page: y
generate lineprinter ripple pattern; sample commad to use it: lptest | lpr -P<printer-name>
```

```
/usr/ucb/lprm        Owner: root      Groups: daemon          permissions: 6711     man page: y
Remove jobs from the line printer spooling queue
/var/spool/*         spool directory per printer
/var/spool/*/lock    Lockfile to obtain pid of current deamon and job number of current job
```

```
/usr/ucb/lpr         Owner: root      Groupd: daemon, lpr     permissions: 6711     man page: y
lpr uses a spooling deamon to print named files
/etc/printcap
/etc/passwd          used for personal (user) identification
/usr/sbin/lpd*       line printer deamons
/var/spool/output/*  directories used for spooling (? isn't it /var/spool/* *=printer ?)
/var/spool/output/*cf* deamon control files
/var/spool/output/*df* data files specified in cf* files
/var/spool/output/*tf* temporary copies of cf* files
```

```
/usr/ucb/lpq        Owner: root      Groups: daemon, lpq     permissions: 6711     man page: y
spool queue examination program
/etc/printcap        used to determine printer characteristics
/var/spool/*         spool directory per printer
/var/spool/*cf*      control files
/var/spool/*/lock    lock file to obtain currently active job
/usr/share/misc/termcap for manipulating the screen for repeted display
```

```
/usr/lib/lpd        Owner: root      Groups: wheel, lpd      permissions: 700      man page: y
line printer spooler deamon
/etc/printcap        printer description file
/etc/hosts.equiv     machine names allowed printer access
/etc/hosts.lpd       machine names allows printer access (under other administrative control)
/var/spool/*         spooldirectoies
/var/spool*/minfree  min free space to leave
/dev/lp*             line printer devices
/dev/printer         socket for local requests
```

```
/usr/etc/lpc        Owner: root      Groups: daemon, lpc     permissions: 2755     man page: y
line printer control program
/etc/printcap        used to determine printer characteristics
/var/spool/*         spool directory per printer
/var/spool/*/lock    lock file for queue control
```

```
/usr/lib/lpf        Owner: root      Groups: wheel, lpf      permissions: 755      man page: n
general printer filter
```

/etc/services must contain the lpd service definition as follows

```
# service name  port/protocol  aliases      # comment
printer         515/tcp        spooler      # remote print spooling
```

```

/etc/syslog.conf      should contain an entry for lpr/lpd, i.e.
lpr.debug            /var/log/lpd-errs

/etc/passwd          lp user with home drectory /var/spool/lpd
lp:*:4:7:lp/var/spool/lpd:/sbin/nologin

/etc/group           lp group with ID 7 must exist
lp:*:7:daemon,lp

/etc/printcap        example entry
HP880C:lf=/var/spool/HP880C/printer-log:sd=/var/spool/HP880C:lp=192.168.99.11:rp=lpt1:mx#0:sf:sh:

```

Findings:

- lpd, lpc, lpq, lprm, lptest work in general
- lpr is able to queue jobs
- files are created successfully in the /var/spool/HP880C directory
- lpr is not able to trigger / communicate with lpd
- lpd is not automatically picking up new print jobs as expected
- when (stopping and) starting lpd, existing queued jobs are printed
- this is normal behaviour since lpd is checking for jobs at start up (to catch up from crashes)
- so still there seems to be a communication problem
- when started lpd creates it's PID and lock file /var/spool/lpd/lpd.lock as expected

??? questions, thoughts, tests ???

- is /dev/printer available? >>> where does it point to? should I ceate it? what should it link to?
- /dev/printer is only used for local printers
- for the remote printer /dev/lp is used as defined in the /ect/printcap
- is there an /usr/sbin/lpd* ? whats needed here?
- no
- ln -s /usr/lib/lpd /usr/sbin/lpd > does not help
- ln -s /usr/lib/lpd /usr/sbin/lpdHP880C > does not help
- no idea if how/if this is used
- which PATH has to be set ?
- /usr/ucb is in the PATH in Mint98
- adding /usr/usb:/usr/lib/usr/etc to the path does not help, but makes using lpr, lprm, lpq, lptest, lpc much more easy
- all above files must be copied ...
- done + permissions set
- portscan
- does not show ANY open ports on my Milan, stange ... 515 should be open?
- how to install man pages into the existing EasyMint system?

Using lpd to print from NVDI:

Mint 98 includes a small program (lpspool) which helps to set this up easily.

NVID must be configured to print to a pipe which is named after the printer defined in /etc/printcap (/pipe/HP880C in my case).

The program lpspool must be running. It picks up the print data from the pipe and trites into the print queue using lpr. This results in to a regulat lpd print job. That's it!

- how to start lpspool?
- manually
- from mint.cnf lik in mint98: exec c:\multititos\bin\sh c:\multititos\bin\lpspool.prg
- I wrote a small script for start / stop / status > this makes it easy to control lpspool and it can be easily included in i.e. std.services