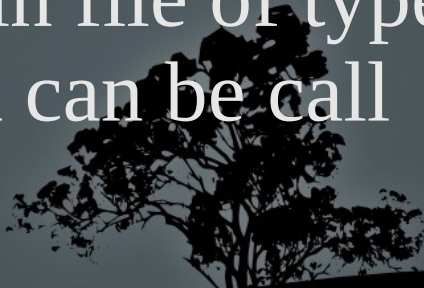


# LPIC-1 101-500 – Lesson 14


## 102.3 Manage shared libraries



# Libraries

- Libraries are resources collections used in software development
  - They can contain code, routines, classes, variables, constants or types
  - They are categorized in static and shared libraries
  - In static libraries the code is embedded in the main program during linking
  - In shared libraries the code resides in file of type **lib\*.so\*** (so = Shared Object) and can be call from many other programs
- 

# Library Paths

- **/lib** # system programs libraries path
  - **/lib32** # location of 32bit libraries in a 64bit system
  - **/lib64** # location of 64bit libraries in a 32bit system
  - **/usr/lib** # user programs libraries path
  - **/usr/local/lib/** # alternative programs libraries path
  - **\$ find / -name "\*.so\*" # find all shared libraries (.so.) in the system**
- 

# Use `ldd` to find the libraries dependencies of programs

- `$ ldd /bin/bash # show bash shared libraries`

```
linux-vdso.so.1 => (0xb773d000)
libncurses.so.5 => /lib/libncurses.so.5 (0xb7703000)
libdl.so.2 => /lib/tls/i686/cmov/libdl.so.2 (0xb76ff000)
libc.so.6 => /lib/tls/i686/cmov/libc.so.6 (0xb75b0000)
/lib/ld-linux.so.2 (0xb773e000)
```

- `$ ldd /usr/bin/top # show top shared libraries`

```
linux-vdso.so.1 => (0x00007fff89ee3000)
libproc-3.2.8.so => /lib/libproc-3.2.8.so
(0x00007fc272a84000)
libncurses.so.5 => /lib/libncurses.so.5
(0x00007fc272863000)
libc.so.6 => /lib/x86_64-linux-gnu/libc.so.6
(0x00007fc2724c3000)
libdl.so.2 => /lib/x86_64-linux-gnu/libdl.so.2
(0x00007fc2722bf000)
libtinfo.so.5 => /lib/libtinfo.so.5 (0x00007fc272098000)
/lib64/ld-linux-x86-64.so.2 (0x00007fc272cce000)
```

# Dynamic Linker *ld.so* and *ld-linux.so*

- `$ ls -l /lib/ld* # show dynamic linker on old systems`
- `$ ls -l /lib64/ld-linux-x86-64.so.2 # on new systems`
- The `ld-<version>.so` or `ld-linux.so.<x>` file is the dynamic linker responsible to find the shared libraries dependencies for all programs
- If a library is missing the program will fail



# Configuration files of Dynamic Linker

- `$ ls -lad /etc/ld.* # show configuration files  
# for Dynamic Linker`  
  
`-rw-r--r-- 1 root root 138216 2011-10-24 14:26  
/etc/ld.so.cache  
-rw-r--r-- 1 root root 34 2011-08-27 07:50  
/etc/ld.so.conf  
drwxr-xr-x 2 root root 4096 2011-10-15 07:06  
/etc/ld.so.conf.d`
- `/etc/ld.so.cache` # binary configuration file. This provides faster access to the configuration than text files
- `/etc/ld.so.conf` # more paths can be added for Dynamic Linker to look for shared libraries. After every change we run `ldconfig` to update `ld.so.cache`
- `/etc/ld.so.conf.d/` # in modern system we don't edit `/etc/ld.so.conf` directly but instead we create `.conf` files under `/etc/ld.so.conf.d/`



# Update the *ld.so.cache* file with `ldconfig`

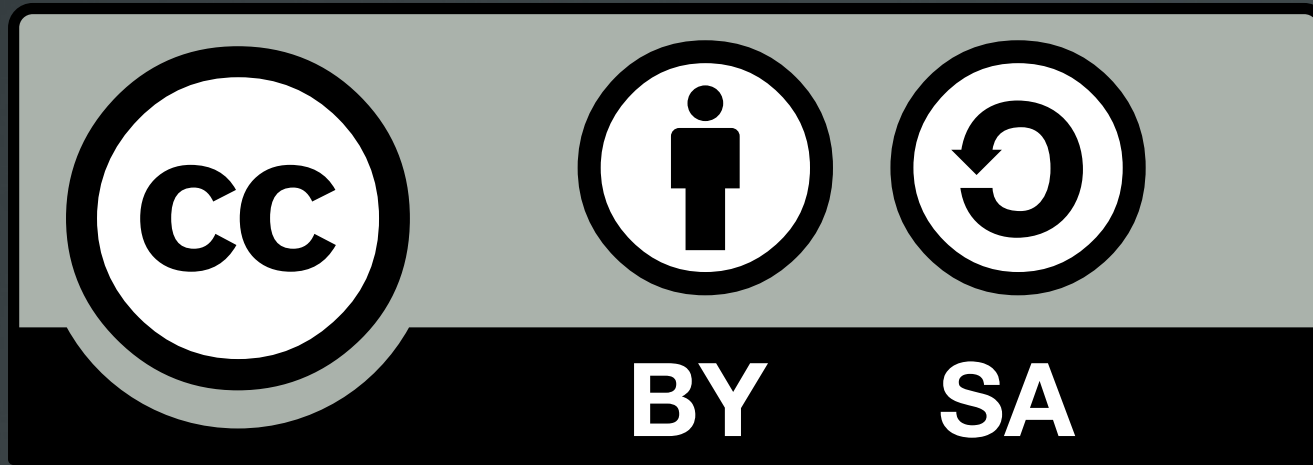
- `$ ldconfig -p` # show the contents of `ld.so.cache`
- `# ldconfig` # read `ld.so.conf` and files under `ld.so.conf.d/` and push changes to `ld.so.cache`

## Options:

- `-p` # show shared libraries included in `ld.so.cache`
- `-v` # verbose view



# License



The work titled "LPIC-1 101-500 – Lesson 14" by Theodotos Andreou is distributed with the Creative Commons Attribution ShareAlike 4.0 International License.

