

pack_libs.sh

Voici un script qui copie toutes les bibliothèques dynamiques nécessaires au bon fonctionnement d'un programme dans un répertoire donné.

```
pack_libs.sh mon_programme ses_libs
```

À vous ensuite de faire un truc du genre :

```
LD_LIBRARY_PATH=ses_libs ./mon_programme
```

pack_libs.sh

```
#!/bin/bash

function syntax () {
    cat <<__EoF__
Syntax: $0 binary_file libdir
    libdir may or may not exist.
__EoF__
exit 1
}

declare -r RED="\e[31;1m"
declare -r GREEN="\e[32;1m"
declare -r YELLOW="\e[33;1m"
declare -r NORM="\e[0m"
declare -i COUNT=0 BAD=0 ERRORS=0
declare DIR BAD LIST_BAD OLD_IFS IFS

if [ "$#" -ne 2 ]; then
    syntax
fi

file "$1" | grep ELF 2>&1 > /dev/null
if [ $? != 0 ]; then
    echo "$1 is not an ELF binary, can't do anything about it..."
    exit 2
else
    if [ ! -d "$2" ]; then
        echo "Creating $2"
        mkdir -p "$2"
    else
        echo "$2 exists"
    fi
    DIR="$2"

    OLD_IFS="$IFS"
```

```

IFS=$'\n'
while read lib; do
  # Bash substitution will get rid of trailing spaces
  cp `readlink -f "${lib%% }"` "$DIR/`basename $lib`"
  if [ $? -eq 0 ];then
    ((COUNT++))
  else
    ((ERROR++))
  fi
done <<( ldd "$1" | cut -d' ' -f3- | cut -d'(' -f1 | grep "^/" )
# Old version couldn't handle spaces in dir name
#done <<( ldd "$1" | awk '{ print $3 }' | grep "^/" )
IFS="$OLDIFS"
LIST_BAD=$(ldd "$1" | grep -i "not found" | cut -d' ' -f1)
BAD=$(echo -n $LIST_BAD | tr ' ' '\n' | wc -l)
fi

echo
echo -e "${GREEN}$COUNT libs${NORM} copied"
[[ $ERROR -ne 0 ]] && echo -e "${YELLOW}$ERROR errors${NORM}"
[[ $BAD -ne 0 ]] && echo -e "${RED}$((BAD+1)) libs${NORM} not found:\n$LIST_BAD"

cat <<__EoF__

```

In order to use this, you should create a script like this:

```

#!/bin/bash

export LD_LIBRARY_PATH=".$DIR":${LD_LIBRARY_PATH}
exec "$1"

#EoF
__EoF__

exit 0

# EoF

```

From:
<http://wiki.free-unices.org/> - Chez moi...

Permanent link:
http://wiki.free-unices.org/doku.php/jeux/pack_libs.sh

Last update: **2015/12/04 11:40**

