

LFS 13.0

Scripts

/etc/profile.d/build-default-flags.sh

```
# Graphite (LTO -fgraphite-identity -floop-nest-optimize -floop-interchange)
COMMON_FLAGS="-march=native -O3 -pipe -flto=auto -ffat-lto-objects -
fgraphite-identity -floop-nest-optimize -floop-interchange"
export LDFLAGS="-flto=auto -Wl,-O1 -Wl,--as-needed"
source /etc/flags-store/export.list
alias clear-flags='unset CFLAGS CXXFLAGS FCFLAGS FFLAGS RUSTFLAGS LDFLAGS'
```

/etc/flags-store/toggle-flags.sh

```
#!/usr/bin/env bash

FLAG_DIR="/etc/flags-store"

GREEN='\033[0;32m'
YELLOW='\033[1;33m'
CYAN='\033[0;36m'
NC='\033[0m'

if [ ! -d "$FLAG_DIR" ]; then
    echo -e "${YELLOW}      :                ($FLAG_DIR)가                .${NC}"
    echo -e "      :                ($FLAG_DIR)                ."
    mkdir -pv $FLAG_DIR
    echo -e "${GREEN}      :                .${NC}"
    echo -e '#Basic C, CXX and LD Flags.
(native,03,pipe)\nCOMMON_FLAGS="-march=native -O3 -pipe"\nexport
CFLAGS="${COMMON_FLAGS}"\nexport CXXFLAGS="${COMMON_FLAGS}"\nexport
FCFLAGS="${COMMON_FLAGS}"\nexport FFLAGS="${COMMON_FLAGS}"\nexport
RUSTFLAGS="-C target-cpu=native -C opt-level=3"\n\nexport LDFLAGS="-Wl,-O1 -
Wl,--as-needed"' | tee ${FLAG_DIR}/safe.flags
    echo -e "${NC}"
fi

mapfile -t FLAG_FILES < <(ls "$FLAG_DIR"/*.flags 2>/dev/null)

echo -e "${CYAN}===== ${NC}"
echo -e "      CFLAGS: ${GREEN}${CFLAGS:- "None"}${NC}"
echo -e "-----"
echo -e "      가                :"
```

```
for i in "${!FLAG_FILES[@]}"; do
    filename=$(basename "${FLAG_FILES[$i]}")
    description=$(head -n 1 "${FLAG_FILES[$i]}" | sed 's/^\#//g' | xargs)
    echo -e "  $((i+1)) ${YELLOW}${filename%-6}${NC} \t-
${CYAN}${description}${NC}"
done
```

```
done

echo -e "  c) ${GREEN}Clear-Flags${NC} (Unset all)"
echo -e "  q) Quit"
echo -e "-----"

read -p "                : " choice

if [[ "$choice" == "q" ]]; then
    echo " ."
elif [[ "$choice" == "c" ]]; then
    unset CFLAGS CXXFLAGS FCFLAGS FFLAGS RUSTFLAGS LDFLAGS MAKEFLAGS
    COMMON_FLAGS
    echo -e "${GREEN}>>>                가                .${NC}"
elif [[ "$choice" =~ ^[0-9]+$ ]] && [ "$choice" -ge 1 ] && [ "$choice" -le
"${#FLAG_FILES[@]}" ]; then
    selected_file="${FLAG_FILES[$((choice-1))]}"
    source "$selected_file"
    echo -e "${GREEN}>>> [(basename "$selected_file")]                !${NC}"
else
    echo -e "${RED}                .${NC}"
fi

unset FLAG_DIR FLAG_FILES

export MAKEFLAGS="-j$(nproc)"
```

/etc/flags-store/export.list

```
export CFLAGS="${COMMON_FLAGS}"
export CXXFLAGS="${COMMON_FLAGS}"
export FCFLAGS="${COMMON_FLAGS}"
export FFLAGS="${COMMON_FLAGS}"
export RUSTFLAGS="-C target-cpu=native -C opt-level=3"
```

/etc/flags-store/01-safe.flag

```
#Basic (-native -O3 - pipe)
COMMON_FLAGS="-march=native -O3 -pipe"
export LDFLAGS="-Wl,-O1 -Wl,--as-needed"
source /etc/flags-store/export.list
```

/etc/flag-store/02-lto.flags

```
#LTO (Basic + -flto=auto -ffat-lto-objects)
COMMON_FLAGS="-march=native -O3 -pipe -flto=auto -ffat-lto-objects"
export LDFLAGS="-flto=auto -Wl,-O1 -Wl,--as-needed"
source /etc/flags-store/export.list
```

/etc/flags-store/03-graphite.flags

```
ln -sv /etc/profile.d/build-default-flags.sh /flags-store/03-graphite.flags
```

```
/etc/flags-store/04-kernel.flags
```

```
#Kernel (-native -O2 - pipe)
COMMON_FLAGS="-march=native -O2 -pipe"
source /etc/flags-store/export.list
```

```
/etc/flags-store/99-debug.flags
```

```
# DEBUG (Graphite -fopt-info)
COMMON_FLAGS="-march=native -O3 -pipe -flto=auto -ffat-lto-objects -
fgraphite-identity -floop-nest-optimize -floop-interchange -fopt-info"
export LDFLAGS="-flto=auto -Wl,-O1 -Wl,--as-needed"
source /etc/flags-store/export.list
```

Optimize Failed Package

- Glibc: LTO-x (safe: -march=native -O3 -pipe)
- TCL: LTO-x (safe: -march=native -O3 -pipe)
- libtool: LTO-x (safe: -march=native -O3 -pipe)
- OpenSSL: LTO-x (safe: -march=native -O3 -pipe)
- Linux-6.18.10: LTO-x, O3-x (safe: -march=native -O2 -pipe)
- frei0r-2.5.1: LTO-x -march=native-x (safe: -O3 -pipe)
- Node.js-22.22.0: LTO-x (safe: -march=native -O3 -pipe) (-shared-sqlite: x)
- SpiderMonkey from firefox-140.8.0: LTO-x (safe: -march=native -O2 -pipe)
- libaio-0.3.113: LTO=x (safe: -march=native -O3 -pipe)
- FFmpeg-8.0.1: LTO=x (safe: -march=native -O3 -pipe)

Chapter 8.30. GCC-15.2.0

```
gcc -v
Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/usr/libexec/gcc/x86_64-pc-linux-gnu/15.2.0/lto-wrapper
Target: x86_64-pc-linux-gnu
Configured with: ../configure --prefix=/usr LD=ld \
--enable-languages=c,c++ --enable-default-pie --enable-default-ssp \
--enable-host-pie --disable-multilib --disable-bootstrap \
--disable-fixincludes --with-system-zlib --with-isl --enable-lto
Thread model: posix
Supported LTO compression algorithms: zlib zstd
gcc version 15.2.0 (GCC)
```

BLFS 13.0

- [BLFS Build Order](#)

Chapter 11. General Utilities

pinentry-1.3.2

```
sed -i "/FLTK 1/s/3/4/" configure &&  
## sed -i '14456 s/1.3/1.4/' configure  
sed -i '14466 's/1.3/1.4/' configure
```

From:

<https://www.gamu.kr/dokuwiki/> -

Permanent link:

<https://www.gamu.kr/dokuwiki/linuxfromscratch/13.0/start?rev=1774927715>

Last update: **2026/03/31 03:28**

