2.2. Требования к хост-системе

2.2.1. Аппаратное обеспечение

Редакторы LFS рекомендуют, чтобы процессор имел не менее четырех ядер и не менее 8 ГБ памяти. Старые системы, не отвечающие этим требованиям, будут по-прежнему работать, но время сборки пакетов будет значительно больше, чем указано в документации.

2.2.2. Программное обеспечение

Ваша хост-система должна иметь следующее программное обеспечение с указанными минимальными версиями. Это не должно быть проблемой для большинства современных дистрибутивов Linux. Также обратите внимание на то, что многие дистрибутивы помещают заголовочные файлы в отдельные пакеты, как правило в формате <package-name>-devel или <package-name>-dev. Обязательно установите эти пакеты, если ваш дистрибутив их предоставляет.

Более ранние версии перечисленных ниже пакетов могут работать, но это не проверялось.

- Bash-3.2 (/bin/sh должен быть символической или жесткой ссылкой на bash)
- Binutils-2.13.1 (Версия выше 2.42 не рекомендуется, так как она не тестировалась)
- Bison-2.7 (/usr/bin/yacc должен быть ссылкой на bison или небольшой скрипт, запускающий bison)
- Coreutils-8.1
- Diffutils-2.8.1
- Findutils-4.2.31
- Gawk-4.0.1 (/usr/bin/awk должен быть ссылкой на gawk)
- GCC-5.2, включая компилятор C++, g++ (версии выше 13.2.0 не рекомендуются, поскольку они не тестировались). Также должны присутствовать стандартные библиотеки C и C++ (с заголовочными файлами), чтобы компилятор C++ мог осуществлять сборку программ.
- Grep-2.5.1a
- Gzip-1.3.12
- Linux Kernel-4.19

Причиной, по которой указаны минимальные требования к версии ядра, является то, что мы указываем эту версию при сборке glibc в Глава 5 и Глава 8. Так как более старые ядра не поддерживаются, скомпилированный пакет glibc немного меньше и быстрее. По состоянию на февраль 2024 г. 4.19 является самой старой версией ядра, поддерживаемой разработчиками ядра. Некоторые версии ядра, более старые, чем 4.19, могут по-прежнему поддерживаться сторонними командами, но они не считаются официальными выпусками ядра; подробности читайте на странице https://kernel.org/category/releases.html

Если версия ядра хоста более ранняя, чем 4.19, вам необходимо обновить ядро на более современную версию. Есть два способа сделать это. Во-первых, посмотрите, предоставляет ли ваш дистрибутив Linux пакет ядра 4.19 или более позднюю версию. Если это так, установите его. Если ваш дистрибутив не предлагает приемлемый пакет ядра или вы предпочитаете не устанавливать его, вы можете скомпилировать ядро самостоятельно. Инструкции по

компиляции ядра и настройке загрузчика (при условии, что хост использует GRUB) находятся в Глава 10.

Для сборки LFS необходимо, чтобы ядро хоста поддерживало псевдотерминал UNIX 98 (РТҮ). Обычно он включен на всех настольных или серверных дистрибутивах, поставляющих Linux 4.19 или более новое ядро. Если на хосте вы используете самостоятельно собранное ядро, убедитесь, что для параметра CONFIG_UNIX98_PTYS установлено значение у в конфигурационном файле ядра.

- M4-1.4.10
- Make-4.0
- Patch-2.5.4
- Perl-5.8.8
- Python-3.4
- Sed-4.1.5
- Tar-1.22
- Texinfo-5.0
- Xz-5.0.0

Важно: Обратите внимание, что упомянутые выше символические ссылки необходимы для создания системы LFS с использованием инструкций, содержащихся в этой книге. Симлинки, указывающие на другое программное обеспечение (например, dash, mawk и т. д.), могут работать, но не тестируются и не поддерживаются командой разработчиков LFS, и могут потребовать либо отклонения от инструкций, либо дополнительных исправлений для некоторых пакетов.

Чтобы узнать, есть ли в вашей хост-системе все необходимые пакеты и возможность компилировать программы, выполните следующий скрипт:

```
cat > version-check.sh << "EOF"
#!/bin/bash
# A script to list version numbers of critical development tools
# If you have tools installed in other directories, adjust PATH here AND
# in ~lfs/.bashrc (section 4.4) as well.

LC_ALL=C
PATH=/usr/bin:/bin

bail() { echo "FATAL: $1"; exit 1; }
grep --version > /dev/null 2> /dev/null || bail "grep does not work"
sed '' /dev/null || bail "sed does not work"
sort /dev/null || bail "sort does not work"

ver_check()
{
   if ! type -p $2 &>/dev/null
```

https://wwoss.ru/ Printed on 2025/11/06 10:55

```
then
     echo "ERROR: Cannot find $2 ($1)"; return 1;
   v=\$(\$2 --version 2>\&1 | grep -E -o '[0-9]+\.[0-9\.]+[a-z]*' | head -n1)
   if printf '%s\n' $3 $v | sort --version-sort --check &>/dev/null
                    %-9s %-6s >= $3\n" "$1" "$v"; return 0;
     printf "OK:
     printf "ERROR: %-9s is TOO OLD ($3 or later required)\n" "$1";
     return 1;
   fi
}
ver_kernel()
{
   kver=\$(uname -r | grep -E -o '^[0-9\.]+')
   if printf '%s\n' $1 $kver | sort --version-sort --check &>/dev/null
   then
     printf "OK:
                    Linux Kernel $kver >= $1\n"; return 0;
   else
     printf "ERROR: Linux Kernel ($kver) is TOO OLD ($1 or later
required)\n" "$kver";
     return 1;
   fi
}
# Coreutils first because --version-sort needs Coreutils >= 7.0
ver check Coreutils
                                   8.1 || bail "Coreutils too old, stop"
                         sort
ver check Bash
                         bash
                                   3.2
ver check Binutils
                                   2.13.1
                         ld
ver check Bison
                         bison
                                   2.7
ver check Diffutils
                                   2.8.1
                         diff
ver check Findutils
                         find
                                   4.2.31
ver check Gawk
                                   4.0.1
                         gawk
                                   5.2
ver check GCC
                         gcc
ver check "GCC (C++)"
                                   5.2
                         q++
                                   2.5.1a
ver check Grep
                         grep
ver_check Gzip
                                   1.3.12
                         gzip
ver check M4
                         m4
                                   1.4.10
ver check Make
                                   4.0
                         make
ver check Patch
                         patch
                                  2.5.4
ver check Perl
                         perl
                                   5.8.8
ver check Python
                         python3 3.4
ver check Sed
                                   4.1.5
                         sed
ver check Tar
                                   1.22
                         tar
ver check Texinfo
                         texi2any 5.0
ver check Xz
                                   5.0.0
                         ΧZ
ver kernel 4.19
if mount | grep -q 'devpts on /dev/pts' && [ -e /dev/ptmx ]
then echo "OK:
                  Linux Kernel supports UNIX 98 PTY";
```

 $\frac{\text{upuace.}}{2025/04/05} software: linux_server: lfs: lfs-12.1: chapter 02: hostreqs \ https://wwoss.ru/doku.php?id=software: linux_server: lfs: lfs-12.1: chapter 02: hostreqs \ https://www.server: lfs-12.1: chapter 02: hostreqs$

```
else echo "ERROR: Linux Kernel does NOT support UNIX 98 PTY"; fi
alias check() {
   if $1 --version 2>&1 | grep -qi $2
   then printf "OK: %-4s is $2\n" "$1";
   else printf "ERROR: %-4s is NOT $2\n" "$1"; fi
}
echo "Aliases:"
alias check awk GNU
alias check yacc Bison
alias check sh Bash
echo "Compiler check:"
if printf "int main(){}" | g++ -x c++ -
then echo "OK:
                 g++ works";
else echo "ERROR: g++ does NOT work"; fi
rm -f a.out
if [ "$(nproc)" = "" ]; then
   echo "ERROR: nproc is not available or it produces empty output"
else
   echo "OK: nproc reports $(nproc) logical cores are available"
fi
E0F
bash version-check.sh
```

← Введение Этапы сборки системы LFS →

https://wwoss.ru/ - worldwide open-source software

Permanent link:

https://wwoss.ru/doku.php?id=software:linux server:lfs:lfs-12.1:chapter02:hostreqs&rev=174383672

Last update: 2025/04/05 10:05



https://wwoss.ru/ Printed on 2025/11/06 10:55