

Тема №6. Особенности сборки ядра Linux

20.10.2021

Общие требования

1. Необходимо написать отчёт о выполненной работе согласно приложенным требованиям.
2. Название лабораторной работы берётся из индивидуального задания.
3. Целью работы является не процесс (сборка, установка и т. п.), а основной конечный результат работы, который нужно достичь. Цель работы указана в задании.
4. Для достижения цели необходимо сформулировать задачи. Они формулируются, исходя из основных проводимых действий.
5. Отчёт состоит из *основных структурных элементов* согласно *требованиям*. В конце заголовком **не ставятся** знаки препинания.
6. *Основная часть* отчёта делится на структурные элементы, **созвучные задачам**.
7. В каждом таком разделе приводится решение поставленной задачи (вызываемые команды, их вывод и т. д.). Снимки экрана *не приводятся*.
8. Отчёт пишется в третьем лице с тем расчётом, что можно понять, как именно была проведена работа (даже с расчётом чтобы можно было её повторить, например).
9. Приводите возникшие трудности, которые преодолевались.
10. В отчёте обязательно приводятся выводы. Абзацы выводов **не нумеруются ни маркированным, ни нумерованным списком**.
11. В выводах как минимум указывается:
 - достигнутая цель;
 - решённые и нерешённые задачи;
 - необходимые показатели, которые были получены, сняты;
 - возникшие трудности;
 - предлагаемые решения и рекомендации;
 - выводы по работе, которые вы сделали для себя.

Индивидуальные задания

1. Объём символьной информация ядра

Цель работы – размер символьной информации ядра после сборки.

Задание – собрать ядро с использованием символьной информации и без неё, определить объём символьной информации, включаемой в ядро.

2. Ускорение повторной сборки ядра с помощью `ccache`

Цель работы – ускорение повторной сборки ядра с помощью `ccache`.

Задание – собрать ядро без использования символьной информации. После собрать ядро повторно (*не очищая* дерево сборки) без использования и с использованием инструмента `ccache`; определить ускорение времени *повторной* сборки за счёт инструмента `ccache`.

3. Ускорение сборки ядра с помощью `tmpfs`

Цель работы – ускорение сборки ядра с помощью `tmpfs`.

Задание – собрать ядро без использования символьной информации без размещения и с размещением всего дерева исходных файлов в файловой системе `tmpfs`; определить ускорение времени сборки за счёт такого размещения исходных файлов.

4. Ускорение сборки ядра за счёт использования каналов

Цель работы – ускорение сборки ядра за счёт использования каналов.

Задание – собрать ядро без использования символьной информации без использования и с использованием каналов вместо временных файлов (см. опцию `pipe` в `GCC`); определить ускорение времени сборки за счёт использования каналов.

5. Наложение заплат дистрибутива на ванильное ядро

Цель работы – наложение заплат дистрибутива на ванильное ядро.

Задание – собрать ядро для вашего дистрибутива, используя исходный код, предоставляемый дистрибутивом, и используя исходный код, предоставляемый ресурсом `kernel.org`, с наложением на него заплат от вашего дистрибутива.

6. Переход под управление другим ядром

Цель работы – переход под управление другим ядром.

Задание – собрать отдельное ядро для вашего дистрибутива, используя `kexec`

7.

–

–

().

8. Наложение заплат на ядро, используя git

Цель работы – ядро с заплатами, собранное с использованием git.

Задание – используя git, скачать исходный код ядра, предоставляемый ресурсом kernel.org, наложить на него заплаты с использованием git, собрать ядро с заплатами.