

ГОУ ВО АЛТАЙСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Физико-технический факультет

Кафедра вычислительной техники и электроники

Вспомогательные программы и команды

(Отчёт по индивидуальному заданию по курсу «ОПЕРАЦИОННЫЕ СИСТЕМЫ»)

Выполнил студент 2-го курса,
585 группы:

_____ А. К. Рожен-
цев

«___» _____ 2020 г.

Проверил

_____ П. Н. Уланов

«___» _____ 2020 г.

Барнаул, 2020 г.

Содержание

1	Введение и постановка задачи	1
2	Теоретическое описание задачи	1
3	Вывод по работе	1

1 Введение и постановка задачи

Ознакомиться с командами `man`, `screen`, `ssh`, `rsync`, `cron`, `make`. Привести краткое описание к каждой команде. Использовал для работы терминал `UXTerm`

2 Теоретическое описание задачи

Краткое описание команд:

`man` — команда предназначенная для форматирования и вывода справочных страниц. Поставляется почти со всеми UNIX-подобными дистрибутивами. Каждая страница справки является самостоятельным документом и пишется разработчиками соответствующего программного обеспечения. (Рис.1-2)

`screen` — команда является консольной программой и имеет однозначное определение, соответствующее названию, — это оконный менеджер, разделяющий один физический терминал между несколькими процессами. (Рис.3)

`ssh` — это протокол удаленного управления компьютером с операционной системой Linux. В основном `ssh` используется для удаленного управления серверами через терминал. (Рис.5)

`rsync` — программа для UNIX-подобных систем, которая выполняет синхронизацию файлов и каталогов в двух местах с минимизированием трафика, используя кодировку данных при необходимости. (Рис.6)

`cron` — это планировщик заданий, который работает на Unix/linux хостинге. Он позволяет автоматически выполнять определенные действия на сервере (запуск программ, скриптов и т. д.), с заданным временем или периодичностью. (Рис.7)

`make` — утилита, автоматизирующая процесс преобразования файлов из одной формы в другую. Чаще всего это компиляция исходного кода в объектные файлы и последующая компоновка в исполняемые файлы или библиотеки. (Рис.8)

```
user@kc203-12:~$ man ls
user@kc203-12:~$ █
```

Рис. 1: Тест команды man(1)

```
LS(1)                                User Commands                                LS(1)

NAME
  ls - list directory contents

SYNOPSIS
  ls [OPTION]... [FILE]...

DESCRIPTION
  List information about the FILES (the current directory by default).
  Sort entries alphabetically if none of -cftuvSUX nor --sort is speci-
  fied.

  Mandatory arguments to long options are mandatory for short options
  too.

  -a, --all
      do not ignore entries starting with .

  -A, --almost-all
      do not list implied . and ..

  --author
  Manual page ls(1) line 1 (press h for help or q to quit)
```

Рис. 2: Тест команды man(2)

3 Вывод по работе

Ознакомился с командами представленными выше. Привел краткое опи-
сание к каждой команде

```

GNU Screen version 4.05.00 (GNU) 10-Dec-16

Copyright (c) 2010 Juergen Weigert, Sadrul Habib Chowdhury
Copyright (c) 2008, 2009 Juergen Weigert, Michael Schroeder, Micah Cowan,
Sadrul Habib Chowdhury
Copyright (c) 1993-2002, 2003, 2005, 2006, 2007 Juergen Weigert, Michael
Schroeder
Copyright (c) 1987 Oliver Laumann

This program is free software; you can redistribute it and/or modify it under
the terms of the GNU General Public License as published by the Free Software
Foundation; either version 3, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT
ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS
FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with
this program (see the file COPYING); if not, see http://www.gnu.org/licenses/,
or contact Free Software Foundation, Inc., 51 Franklin Street, Fifth Floor,

[Press Space for next page; Return to end.]

```

Рис. 3: Тест команды screen

```

user@kc203-12:~$ ssh
usage: ssh [-1246AaCfGgKkMNnqsTtVvXxYy] [-b bind_address] [-c cipher_spec]
          [-D [bind_address:]port] [-E log_file] [-e escape_char]
          [-F configfile] [-I pkcs11] [-i identity_file]
          [-J [user@]host[:port]] [-L address] [-l login_name] [-m mac_spec]
          [-O ctl_cmd] [-o option] [-p port] [-Q query_option] [-R address]
          [-S ctl_path] [-W host:port] [-w local_tun[:remote_tun]]
          [user@]hostname [command]
user@kc203-12:~$

```

Рис. 4: Тест команды ssh

```

sent 18 bytes  received 12 bytes  60.00 bytes/sec
total size is 0  speedup is 0.00
rsync error: some files/attrs were not transferred (see previous errors) (code 2
3) at main.c(1196) [sender=3,1,2]
user@kc203-12:~/Рабочий стол$ rsync --progress -av dada
sending incremental file list
-rw-r--r--          36 2020/03/05 12:26:26 dada

```

Рис. 5: Тест команды rsync

```

# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/

```

Рис. 6: Тест команды cron

```
user@kc203-12:~$ make
make: Циклическая зависимость main_moncar.o <- main_moncar.o пропущена.
cc -c main_moncar.c
cc -o edit main_moncar.o
user@kc203-12:~$ █
```

Рис. 7: Тест команды make