



10:17:04 / RUNNING

**Problem up02-1-mz02-1 (дореш)**

Open date: 2022/09/13 20:00:00

Deadline: 2022/10/05 10:00:00

mz01-1
mz01-2
mz01-3
mz01-4
mz01-5
mz01-6
up01-1
up01-2
up01-3
up01-4
up01-5
up01-6
mz02-1
mz02-2
mz02-3
mz02-4
mz02-5
mz02-6
up02-1
up02-2
up02-3
up02-4
up02-5
up02-6
up02-7
mz03-1
mz03-2
mz03-3
mz03-4
mz03-5
up03-1
up03-2
up03-3
up03-4
up03-5
mz04-1
mz04-2
mz04-3
mz04-4
mz04-5
mz04-6
up04-1
up04-2
up04-3
up04-4
up04-5
mz05-1
mz05-2
mz05-3
mz05-4
mz05-5
up05-1
up05-2
up05-3
up05-4
up05-5
mz06-1
mz06-2
mz06-3
mz06-4
mz06-5
up06-1
up06-2
up06-3
up06-4



10:17:27 / RUNNING

**Problem up02-2-mz02-2 (дореш)**

Open date: 2022/09/13 20:00:00

Deadline: 2022/10/05 10:00:00

mz01-1
mz01-2
mz01-3
mz01-4
mz01-5
mz01-6
up01-1
up01-2
up01-3
up01-4
up01-5
up01-6
mz02-1
mz02-2
mz02-3
mz02-4
mz02-5
mz02-6
up02-1
up02-2
up02-3
up02-4
up02-5
up02-6
up02-7
mz03-1
mz03-2
mz03-3
mz03-4
mz03-5
up03-1
up03-2
up03-3
up03-4
up03-5
mz04-1
mz04-2
mz04-3
mz04-4
mz04-5
mz04-6
up04-1
up04-2
up04-3
up04-4
up04-5
mz05-1
mz05-2
mz05-3
mz05-4
mz05-5
up05-1
up05-2
up05-3
up05-4
up05-5
mz06-1
mz06-2
mz06-3
mz06-4
mz06-5
up06-1
up06-2
up06-3
up06-4

**Problem up02-3-mz02-3 (дореш)**

Open date: 2022/09/13 20:00:00

Deadline: 2022/10/05 10:00:00

**Problem up02-3: mz02-3 (дореш)**

Задание должно выполняться на сервере, адрес которого вы получили в первом задании.

На сервере HOST в вашем домашнем каталоге находится файл LOGIN.pem, где LOGIN - ваш логин. Этот файл содержит приватный ECDSA ключ. Скопируйте его к себе на локальный компьютер, он потребуется вам в дальнейшем. Приватный ключ нельзя передавать другим людям.

**Подключение к ejudge**

Создайте в домашнем каталоге каталог contest.

```
cd  
mkdir contest
```

Подключите ejudge к этому каталогу.

```
/opt/ejudge/bin/ejudge-fuse --user EJUDGE_LOGIN --url https://unicorn.ejudge.ru/cgi-bin/~contest -o use_ino
```

Здесь EJUDGE\_LOGIN --- Ваш логин от ejudge.

На терминал будет выведено приглашение ко вводу пароля, введите Ваш пароль от ejudge, нажмите ENTER.

Узнайте номер турнира вашей группы в системе ejudge.

```
ls contest
```

На стандартный поток вывода будет выведена строка, аналогичная

```
'461,201 группа, 3 семестр, 2021-2022'
```

Здесь 461 - номер турнира, который далее обозначим CONTEST\_ID.

После этого команда

```
cat contest/CONTEST_ID/problems/PROBLEM/statement.html
```

должна вывести на стандартный поток вывода HTML-текст этого задания.

PROBLEM – это код данной задачи, например, mz02-3 или up02-3.

**Отчет о выполнении задания**

С помощью вашего приватного ключа подпишите файл cat

contest/CONTEST\_ID/problems/PROBLEM/info.json.

Для этого выполните следующие команды:

```
cp contest/CONTEST_ID/problems/PROBLEM/info.json report.txt  
openssl dgst -sha256 -sign LOGIN.pem -hex < report.txt > sig.txt  
cat report.txt sig.txt > report-signed.txt
```

Сдайте файл на проверку в тестирующую систему:

```
cp report-signed.txt contest/CONTEST_ID/problems/PROBLEM/submit/0
```

**Внимание! Эту и все следующие задачи контрольной работы вам будут доступны на сдачу только с вашего сервера.**

mz01-1
mz01-2
mz01-3
mz01-4
mz01-5
mz01-6
up01-1
up01-2
up01-3
up01-4
up01-5
up01-6
mz02-1
mz02-2
mz02-3
mz02-4
mz02-5
mz02-6
up02-1
up02-2
up02-3
up02-4
up02-5
up02-6
up02-7
mz03-1
mz03-2
mz03-3
mz03-4
mz03-5
up03-1
up03-2
up03-3
up03-4
up03-5
mz04-1
mz04-2
mz04-3
mz04-4
mz04-5
mz04-6
up04-1
up04-2
up04-3
up04-4
up04-5
mz05-1
mz05-2
mz05-3
mz05-4
mz05-5
up05-1
up05-2
up05-3
up05-4
up05-5
mz06-1
mz06-2
mz06-3
mz06-4
mz06-5
up06-1
up06-2
up06-3
up06-4

**Problem up02-4-mz02-4 (дореш)**

Open date: 2022/09/13 20:00:00

Deadline: 2022/10/05 10:00:00

**Problem up02-4: mz02-4 (дореш)****Практическое занятие «Введение в UNIX», задача 3**

Задание должно выполняться на сервере HOST

При выполнении этого практического занятия вам необходимо проделать все шаги, описанные ниже и дать ответы на все вопросы. Ответы на все вопросы должны находиться в одном текстовом файле, который вы сладите в качестве отчета о выполнении занятия. Файл с ответами на вопросы должен иметь следующую структуру:

```
ANSWERS
**Answer 1:
текст ответа на вопрос 1
**Answer 2:
текст ответа на вопрос 2
```

и т.д.

Для редактирования файла с ответами воспользуйтесь консольным текстовым редактором ( nano, pico, vim, mcedit, emacs... ). Консольные текстовые редакторы не создают нового окна графической среды, а работают в окне эмультора терминала, поэтому их можно использовать и тогда, когда графическая оконная среда недоступна, например, при использовании ssh для входа на другой компьютер в сети.

Например, откройте файл на редактирование с помощью редактора nano с помощью команды:

```
sudo nano report.txt
```

в этот файл вы будете записывать ответы на задания. В первой строке текста напишите слово ANSWERS. Для выхода из редактора нажмите Ctrl-X. Подтвердите, что желаете выйти из редактора и подтвердите им сохраняемого файла.

По желанию вы можете использовать редактор vim или редактор emacs в консольном режиме или любой другой консольный редактор.

Снова откройте на редактирование файл answers.txt. Держите его открытым все время, но не забывайте регулярно сохранять файл с помощью Ctrl-O.

Шаг 1. Зайдите на сервер HOST и на нем выполняйте команды. Команда командного процессора имеет следующий общий вид:

```
COMMAND ARGS
```

ARGS – это последовательность аргументов команды. Аргументы делятся на «опции», которые управляют работой программы, и имена файлов. Как правило, опции записываются перед именами файлов (но бывают и исключения). Например, у команды

```
ls -la /etc
```

Аргумент -la – опция (точнее, комбинация двух опций -l и -a), аргумент /etc – имя файла (точнее, каталога).

Шаг 2. Для получения описания команд и программ операционной системы UNIX используется команда man. Например, команда

```
man man
```

выдаст описание самой команды man

```
man ls
```

выдаст описание команды ls (получение списка файлов).

Описание команды может выводиться в постраничном режиме. Для прокрутки страницы вниз можно использовать «пробел» или PgDn, для прокрутки страницы вверх – PgUp. Для выхода из прокрутки используется 'q'.

Вопрос 1. Какая опция команды man эквивалентна команде whatis.

Вопрос 2. Чем отличается man 1 write от man 2 write?

Вопрос 3. Опишите кратко, что делает каждая из команд:

```
cat
grep
head
yes
```

Шаг 3. Для получения списка файлов в каталоге используется команда ls. Чтобы получить список файлов с расширенной информацией о файлах используется опция -l, например:

```
ls -l /usr/bin
```

Вопрос 4. Каким образом упорядочены строки описания файлов, выводимые командой ls?

Вопрос 5. При использовании опции -l на стандартный поток вывода выдается информация о файле в виде:

```
dr-xr-xr-x. 2 root root 20480 Jul 24 11:41 sbin
```

что означает символ 'd' в первой позиции? Какие еще символы могут здесь быть? Что означает 'Jul 24 11:41'?

Вопрос 6. Какие опции позволяют получить список всех файлов с расширенной информацией, упорядоченный в порядке убывания размера?

Шаг 4. Для вывода содержимого файла на стандартный поток вывода, объединения содержимого нескольких файлов, и т. д. используется команда cat. Например,

```
cat /etc/hosts /etc/fstab
```

сначала выводят на стандартный поток вывода содержимое файла /etc/hosts, затем выводят на стандартный поток вывода содержимое файла /etc/fstab.

Вопрос 7. С помощью команды cat выведите содержимое самого маленького регулярного файла (не символьической ссылки), размер которого больше 5 байт в каталоге /etc, к которому [файл] вы имеете доступ на чтение. Укажите имя этого файла.

Шаг 5. Вызов программы можно перенаправить в файл. Для этого предназначены конструкции перенаправления:

> FILE	перенаправление стандартного вывода программы в файл FILE (в режиме перезаписи)
>> FILE	перенаправление стандартного вывода программы в файл FILE (в режиме добавления)
>>> FILE	перенаправление стандартного потока ошибок в файл FILE (в режиме перезаписи)
2> FILE	перенаправление стандартного потока ошибок в файл FILE (в режиме добавления)
2>&1	перенаправление стандартного потока ошибок туда, куда уже перенаправлен стандартный поток вывода
< FILE	перенаправление стандартного потока ввода на чтение из файла FILE

Конструкции перенаправления не считаются аргументами программы и не передаются в программу в качестве аргументов командной строки. В путях файлов, задаваемых в командной строке, символ ~ обозначает домашний каталог текущего пользователя (то есть вас). Например, путь ~/bash\_profile указывает на файл .bash\_profile, расположенный в вашем домашнем каталоге. Например,

```
cat /etc/passwd > ~/file1
```

содержимое файла /etc/passwd выводится на стандартный поток вывода, который перенаправлен в файл ~/file1, то есть содержимое файла /etc/passwd копируется в файл ~/file1. Команда

```
ls -l > ~/file2 2>&1 /etc/passwd /etc/networks
```

перенаправит и стандартный поток вывода, и стандартный поток ошибок в файл ~/file2. При этом команда ls будут переданы следующие аргументы командной строки:

```
ls -l /etc/passwd /etc/networks
```

Вопрос 8. Каково будет содержимое файла ~/file2 после выполнения команды

```
ls -l > ~/file2 2>&1 /etc/passwd /etc/networks
```

Вопрос 9. Запишите несколько вызовов команды cat только с одним аргументом и одним перенаправлением, которые дали бы такой же результат, как и команда

```
cat /etc/hosts /etc/fstab > ~/file3
```

Шаг 6. Права доступа к файлам определяют допустимые операции, которые может выполнить данный пользователь над данным файлом. В Unix системах пользователи делятся на три категории по отношению к данному файлу: пользователь-владелец файла, пользователи, у которых один из идентификаторов группы совпадает с группой файла (другими словами группа-владелец файла), все прочие пользователи. Для каждой из категорий пользователей определяется доступность трех операций: «чтение», «запись» и «исполнение». Каждая операция либо разрешена, либо запрещена. Таким образом для записи прав доступа к файлу достаточно 9 бит (три категории в каждой по три операции). Традиционно для записи прав доступа в числовую форму используется восьмеричными цифрами. Старшая из трех цифр отвечает за права доступа для владельца файла, средняя – для группы файлов, и младшая – для прочих пользователей. В каждом из восьмеричных разрядов бит 4 отвечает за право на чтение, бит 2 – за право на запись, бит 1 – за право на исполнение.

Например, права доступа 641 разрешают чтение и запись для владельца файла, только чтение для группы пользователей файла и только исполнение для всех прочих пользователей.

Подробнее о правах доступа можете прочитать в [Википедии](#).

При выводе списка файлов на экран для записи прав доступа используется символьская форма, в которой знак '-' (минус) обозначает, что соответствующее право отсутствует, 'r' – что есть право на чтение, 'w' – что есть право на запись, 'x' – что есть право на выполнение. Права доступа 641 записываются как rw-r--r--.

Вопрос 10. Для прав доступа в символьической форме запишите права доступа в виде восьмеричного числа и наоборот. В каждой строке сначала записываются права в символьической форме, затем число.

```
rw-r--r-- 641
r--r--r-- ???
?????????? 750
r--r--r--x ????
?????????? 662
```

Шаг 7. При создании файлов необходимо некоторым образом задавать права доступа к создаваемому файлу. Поскольку конструкции перенаправления потоков не позволяют задавать права доступа явно, используются права по умолчанию (для обычных файлов - 666, для каталогов и исполняемых файлов - 777), к которым применяется значение параметра umask. umask задает, какие биты прав доступа нужно удалить («исключить») в правах доступа по умолчанию при создании файла. Например, если права доступа на создаваемый файл, по умолчанию 666 (то есть rw-rw-rw), и параметр umask равен 7, то результатирующие права доступа к созданному файлу будут равны 660 (rw-rw--). Получить текущее значение параметра umask можно с помощью команды umask (без аргументов), а установить – с помощью аргумента команды umask.

Вопрос 11. Какое значение umask у вас установлено? Как после применения umask изменяются права доступа 666 (rw-rw-rw)?

Вопрос 12. Какое значение umask должно быть установлено, чтобы при создании файла с помощью команды

```
echo hello >hell.txt
```

он получил права rw-r--? Запишите команду, которая установит требуемое значение umask.

Шаг 8. Для смены прав доступа к файлам и каталогам используется команда chmod. Например,

```
chmod 700 newtask1
```

изменяет права доступа к каталогу newtask1 на восьмеричное значение 700.

Вопрос 13. Выполните изменение прав доступа к каталогу newtask1 так, чтобы права доступа описывались строкой rwxr--. Какова будет строка вывода команды ls -l, соответствующая каталогу newtask1?

Шаг 9. Для прерывания работы программы используется комбинация Ctrl-C. Команда cat без входных файлов считывает данные со стандартного потока ввода.

Вопрос 14. запустите команду

```
cat > ~/file4
```

затем введите

```
abc
```

и нажмите Enter, затем введите

```
def
```

и нажмите Ctrl-C без нажатия Enter. Каков будет размер файла ~/file4?

Вопрос 15. Почему размер файла из вопроса 14 будет таким

Шаг 10. При вводе данных со стандартного потока ввода для завершения ввода используется комбинация Ctrl-D. Если курсор находится не в первой колонке терминала, Ctrl-D надо нажать дважды.

Вопрос 16. запустите команду

```
cat > ~/file5
```

введите

```
1234
```

нажмите Enter и Ctrl-D, запустите команду

```
cat > ~/file6
```

введите

```
1234
```

и, не нажимая Enter, нажмите Ctrl-D два раза. Каковы размеры файлов ~/file5 и ~/file6? Почему?

Шаг 11. Стандартный ввод программы можно перенаправить на чтение из файла. Для этого предназначена конструкция перенаправления &lt;. Конструкции перенаправления не считаются аргументами программы и не передаются в программу в качестве аргументов командной строки.Например:

```
cat < /etc/passwd >/file7
```

перенаправит стандартный ввод программы cat на чтение из файла /etc/passwd, а стандартный вывод на запись в файл ~/file7. Команда cat без аргументов считывает со стандартного потока ввода, таким образом, данная команда скопирует содержимое файла /etc/passwd в файл ~/file7.

Отчет о выполнении задания

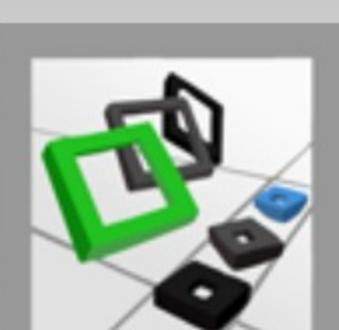
Перед сдачей файла report.txt в тестирующую систему выполните следующую команду:

```
openssl dgst -sha256 -sign LOGIN.pem -hex < report.txt > sig.txt
```

Здесь LOGIN.pem - приватный ключ (см. предыдущие задачи). В результате в конец файла report.txt будет дописана цифровая подпись.

В тестирующую систему сдайте подписанный вами файл report.txt

mz01-1
mz01-2
mz01-3
mz01-4
mz01-5
mz01-6
up01-1
up01-2
up01-3
up01-4
up01-5
up01-6
mz02-1
mz02-2
mz02-3
mz02-4
mz02-5
mz02-6
up02-1
up02-2
up02-3
up02-4
up02-5
up02-6
up03-1
mz03-2
mz03-3
mz03-4
up03-1
up03-2
up03-3
up03-4
up03-5
mz04-1
mz04-2
mz04-3
mz04-4
mz04-5
mz04-6
up04-1
up04-2
up04-3
up04-4
up04-5
mz05-1
mz05-2
mz05-3
mz05-4
mz05-5
up05-1
up05-2
up05-3
up05-4
up05-5
mz06-1
mz06-2
mz06-3
mz06-4
up06-1
up06-2
up06-3
up06-4

**Problem up02-5-mz02-5 (дореш)**

Open date: 2022/09/13 20:00:00

Deadline: 2022/10/05 10:00:00

**Problem up02-5: mz02-5 (дореш)**

Задание должно выполняться на сервере HOST

Правила оформления ответов описаны в предыдущем задании.

Шаг 12. У каждого пользователя есть его личный каталог для хранения файлов в системе. Обычно, этот каталог находится в каталоге /home. Личный каталог пользователя называется «домашним» каталогом. Для смены текущего каталога используется команда cd. Команда cd без аргументов устанавливает домашний каталог в качестве текущего. Например,

```
cd /tmp
```

переход в каталог /tmp

```
cd
```

переход в домашний каталог Для получения полного пути к текущему каталогу используется команда

```
pwd
```

**Вопрос 17. Выполните команды**

```
cd
```

```
pwd
```

что было напечатано на стандартный поток вывода? Что это за путь?

**Вопрос 18. Выполните команду**

```
cd /usr/local/bin/../../../../include/sys/..
```

какой каталог будет установлен в качестве текущего?

Шаг 13. Для создания каталогов используется команда mkdir.

**Вопрос 19. Выполните команды**

```
cd
```

```
mkdir newtask1
```

Каково будет время последней модификации каталога newtask1? Каковы права доступа к этому каталогу (восьмеричные)?

Шаг 14. Для удаления отдельных файлов используется команда rm. Например,

```
rm ~/file2
```

Шаг 15. Для удаления пустого каталога используется команда rmdir. Например,

```
rmdir newtask1
```

Шаг 16. Для удаления файлов и каталогов со всем их содержимым используется команда rm с опцией -r, например,

```
rm -rf .mozilla
```

**Внимание! Используйте с осторожностью!!!**

Шаг 17. Команда echo выводит на стандартный поток вывода свои аргументы командной строки. Например,

```
echo Hello world
```

**Вопрос 20. Выполните команду**

```
echo 1 ~ 2 ~ 3 > ~/file7
```

каков получился размер файла ~/file7? Каково его содержимое?

Шаг 18. В командной строке допускается использовать шаблоны имен файлов. В шаблонах имен файлов символ '\*' (звездочка) означает произвольное количество (в том числе 0) произвольных символов, а символ '?' (вопрос) означает ровно один произвольный символ. Шаблоны имен файлов обрабатываются командным процессором bash перед вызовом запускаемой программы, и поэтому запускаемая программа получает в качестве аргументов командной строки все файлы, удовлетворяющие шаблону, а не сам шаблон.

**Вопрос 21. Выполните команду**

```
echo /etc/c?o*
```

что выведено на стандартный поток вывода?

Шаг 19. Для копирования файлов используется команда cp. Например,

```
cp /etc/passwd ~/file9
```

скопирует файл /etc/passwd в файл ~/file9

```
cp /etc/hosts /etc/group ~
```

скопирует файлы /etc/hosts и /etc/group в каталог /tmp под своими именами (hosts, group)

```
cp /etc/p* ~
```

скопирует все файлы, удовлетворяющие шаблону /etc/p\* в каталог /tmp

Шаг 20. В домашнем каталоге создайте каталог newtask2, затем скопируйте в него файлы /etc/passwd и /etc/hosts. Что будет выведено командой ls -l при распечатке списка файлов в каталоге newtask2?

Шаг 20. Для переноса (переименования) файлов используется команда mv, использование которой аналогично использованию команды cp.

Шаг 23. В каталоге newtask2 переименуйте файл hosts в hosts2. Что будет выведено командой ls -l при распечатке списка файлов в каталоге newtask2?

Шаг 21. Переменные окружения – это способ хранения дополнительной информации (как правило, о настройках пользователя), доступной каждой выполняющейся программе. Переменная окружения ставит соответствие некоторому имени произвольное текстовое значение. Например, переменная окружения HOME задает путь к домашнему каталогу. Чтобы получить список всех установленных переменных окружения используется команда set без аргументов.

Шаг 24. Выполните команду set для просмотра всех переменных окружения. Какое назначение у переменных окружения USER и UID?

Шаг 22. Вывести значение некоторой конкретной переменной окружения можно с помощью команды

```
echo $VAR
```

где VAR – имя переменной окружения. Например,

```
echo $USER
```

Конструкция \$VAR в аргументах командной строки обрабатывается командным процессором bash, поэтому запускаемая программа вместо записи \$VAR получает в качестве аргумента командной строки значение переменной окружения \$VAR.

Шаг 25. Каково значение переменной окружения SHELL?

Шаг 23. Установить или изменить значение переменной окружения можно с помощью команды

```
VAR=value
```

здесь VAR – имя переменной окружения, value – значение переменной окружения. value не должно содержать пробелы. Если value содержит пробелы (которые могут появиться и из-за использования значений других переменных окружения), необходимо value заключить в кавычки:

```
VAR="value"
```

В строке value могут использоваться имена переменных, например,

```
WORKDIR=$HOME/newtask2
```

**Вопрос 26. Выполните команду**

```
PATH="/etc:$PATH:/usr/include"
```

каково будет значение переменной окружения PATH?

Шаг 24. Переменная окружения PATH задает список каталогов, которые просматриваются, если запускается команда, в имени которой отсутствует символ '/'. Каталоги в переменной окружения PATH разделяются символом ':' (двоеточие). Например, при выполнении команды

```
ls /etc
```

для поиска программы (исполняемого файла) ls будет использоваться переменная окружения PATH. При выполнении команды

```
/bin/ls /etc
```

всегда будет использоваться программа /bin/ls независимо от значения переменной окружения PATH.

Шаг 27. В каком каталоге из списка каталогов в переменной окружения PATH находится программа who?

Шаг 25. Для получения кода завершения процесса используется специальная переменная \$? . Если процесс завершился успешно, код завершения равен 0, если процесс завершился с ошибкой, код завершения не 0.

Шаг 28. Запустите ls /123 и ls /home, с какими кодами возврата завершился каждый из процессов? используйте команду echo.

**Отчет о выполнении задания**

Перед сдачей файла report.txt в тестирующую систему выполните следующую команду:

```
openssl dgst -sha256 -sign LOGIN.pem -hex < report.txt > sig.txt
```

Здесь LOGIN.pem - приватный ключ (см. предыдущие задачи). В результате в конец файла report.txt будет дописана цифровая подпись.

В тестирующую систему сдайте подписанный вами файл report.txt

mz01-1
mz01-2
mz01-3
mz01-4
mz01-5
mz01-6
up01-1
up01-2
up01-3
up01-4
up01-5
up01-6
mz02-1
mz02-2
mz02-3
mz02-4
mz02-5
mz02-6
up02-1
up02-2
up02-3
up02-4
up02-5
up02-6
up02-7
mz03-1
mz03-2
mz03-3
mz03-4
mz03-5
up03-1
up03-2
up03-3
up03-4
up03-5
mz04-1
mz04-2
mz04-3
mz04-4
mz04-5
mz04-6
up04-1
up04-2
up04-3
up04-4
up04-5
mz05-1
mz05-2
mz05-3
mz05-4
mz05-5
up05-1
up05-2
up05-3
up05-4
up05-5
mz06-1
mz06-2
mz06-3
mz06-4
mz06-5
up06-1
up06-2
up06-3
up06-4

**Problem up02-6-mz02-6 (дореш)****Open date:** 2022/09/13 20:00:00**Deadline:** 2022/10/05 10:00:00**Problem up02-6: mz02-6 (дореш)****Практическое занятие «Введение в UNIX», задача 5**

Задание должно выполняться на сервере HOST.

Правила оформления ответов описаны ранее.

**Шаг 26.** Для компиляции программ на языке Си используется команда gcc, которая имеет вид

```
gcc OPTIONS-AND-FILES.
```

Например,

```
gcc prog.c -Wall -Werror -std=gnu11 -g -o prog
```

запустит компиляцию программы prog.c. В случае отсутствия ошибок будет создан исполняемый файл prog.

**Вопрос 29.** В домашнем каталоге создайте каталог newtask3. Сделайте этот каталог текущим. В этом каталоге с помощью команды cat создайте файл aplusb.c следующего содержания:

```
void main()
{
int a, b, c;
scanf("%d%d", &a, &b);
c = a + b;
printf("%d\n", c);
}
```

Запустите компиляцию файла командой

```
gcc -Wall -Werror -std=gnu11 -O2 aplusb.c -o aplusb
```

Какие сообщения об ошибках будут напечатаны командой gcc?

**Вопрос 30.** Исправьте ошибки. Добейтесь, чтобы компиляция файла проходила без ошибок. Какую информацию печатает команда ls -l о файле aplusb?**Шаг 27.** Для запуска программы из текущего каталога необходимо использовать явную запись текущего каталога '.', так как текущий каталог не находится в списке каталогов в переменной окружения PATH. Например,

```
./aplusb
```

**Вопрос 31.** Запустите программу aplusb из текущего каталога. В качестве первого числа введите ваш идентификатор пользователя, в качестве второго числа введите размер файла aplusb. Какой будет напечатан результат?**Шаг 28.** В текущем каталоге создайте текстовый файл 001.dat, содержащий два целых числа. Первое число было напечатано командой aplusb в вопросе 26. Второе число – размер файла /etc/passwd. Не забудьте разделить их пробелом или переводом строки.**Вопрос 32.** Какая командная строка необходима для запуска программы aplusb, перенаправления ей на стандартный ввод файла 001.dat и перенаправления ее стандартного вывода в файл 001.res?**Вопрос 33.** Выполните эту команду. С помощью команды cat выведите содержимое файла 001.res.**Шаг 29.** Для измерения времени работы программы используется команда time. Например,

```
time ./aplusb < 001.dat
```

Время работы программы выводится в строке "User".

**Вопрос 34.** Чему равно это время (User)?**Шаг 30.** В программу вставьте цикл задержки вида

```
for (int i = 0; i < N; ++i) {}
```

где константу N подберите таким образом, чтобы время работы программы (User) попало в промежуток от 1.000 до 1.100 секунды. При компиляции программы не используйте опции оптимизации (-O2)!

**Вопрос 35.** Чему равно значение N?**Шаг 31.** Для перенаправления стандартного выходного потока одного процесса на стандартный входной поток другого процесса в командном интерпретаторе используется символ | (вертикальная черта).**Вопрос 36.** Напишите и объясните, что выведет на экран команда:

```
ps -A | grep sh
```

**Шаг 32.** Вопросы повышенной сложности**Вопрос 37.** Что выводит ulimit -c, что делает ulimit -c 0?**Вопрос 38.** Напишите программу, вызывающую segmentation fault, получите core dump-файл. Ответом на вопрос будет являться вывод

```
readelf -S <имя файла>
```

**Вопрос 39.** Напишите команду для командного интерпретатора, выводящую на экран полный листинг (результат работы ls -l) для 5 файлов с самыми маленькими длинами из каталога /etc, в именах которых содержится строка ".conf", отсортированных по возрастанию размера файлов.**Вопрос 40.** Напишите команду для командного интерпретатора, подсчитывающую число слов в файле /usr/share/dict/words, которые не содержат ни одну из латинских букв: a, e, i, o или u.**Отчет о выполнении задания**

Перед сдачей файла report.txt в тестирующую систему выполните следующую команду:

```
openssl dgst -sha256 -sign LOGIN.pem -hex < report.txt > sig.txt
cat sig.txt >> report.txt
```

Здесь LOGIN.pem - приватный ключ (см. предыдущие задачи). В результате в конец файла report.txt будет дописана цифровая подпись.

В тестирующую систему сдайте подписанный вами файл report.txt

mz01-1

mz01-2

mz01-3

mz01-4

mz01-5

mz01-6

up01-1

up01-2

up01-3

up01-4

up01-5

up01-6

mz02-1

mz02-2

mz02-3

mz02-4

mz02-5

mz02-6

up02-1

up02-2

up02-3

up02-4

up02-5

up02-6

up02-7

mz03-1

mz03-2

mz03-3

mz03-4

mz03-5

up03-1

up03-2

up03-3

up03-4

up03-5

mz04-1

mz04-2

mz04-3

mz04-4

mz04-5

mz04-6

up04-1

up04-2

up04-3

up04-4

up04-5

mz05-1

mz05-2

mz05-3

mz05-4

mz05-5

up05-1

up05-2

up05-3

up05-4

mz06-1

mz06-2

mz06-3

mz06-4

mz06-5

up06-1

up06-2

up06-3

up06-4

up06-5

**Problem up02-7-mz02-7 (дореш)**

Open date: 2022/09/13 20:00:00

Deadline: 2022/10/05 10:00:00

**Problem up02-7: mz02-7 (дореш)**В этом домашнем задании вам требуется пройти игру **Bandit** (<https://overthewire.org/wargames/bandit/>).Перед началом установите себе **asciinema** (<https://asciinema.org/docs/installation#installing-on-linux>) и запустите запись действий в терминале с помощью

```
asciinema rec --stdin shell.cast
```

Ответы сохраняйте в файле answers.txt в формате "логин пароль" для каждого уровня в отдельной строчке, начиная с логина bandit0.

По завершению работы остановите запись, отправьте в ejudge тгз-архив с двумя файлами (answers.txt и shell.cast)

mz01-1
mz01-2
mz01-3
mz01-4
mz01-5
mz01-6
up01-1
up01-2
up01-3
up01-4
up01-5
up01-6
mz02-1
mz02-2
mz02-3
mz02-4
mz02-5
mz02-6
up02-1
up02-2
up02-3
up02-4
up02-5
up02-6
up02-7
mz03-1
mz03-2
mz03-3
mz03-4
mz03-5
up03-1
up03-2
up03-3
up03-4
up03-5
mz04-1
mz04-2
mz04-3
mz04-4
mz04-5
mz04-6
up04-1
up04-2
up04-3
up04-4
up04-5
mz05-1
mz05-2
mz05-3
mz05-4
mz05-5
up05-1
up05-2
up05-3
up05-4
up05-5
mz06-1
mz06-2
mz06-3
mz06-4
mz06-5
up06-1
up06-2
up06-3
up06-4