Сессионное задание

[Задача 1. Шерлок Холмс и LVM+RAID 2](#_Toc210587934)

[Постановка задачи 2](#_Toc210587935)

[Объяснение решения 2](#_Toc210587936)

[Решение 3](#_Toc210587937)

[Аппаратные сбои 5](#_Toc210587938)

[Аппаратный сбой 1. Система на первом RAID0 (SSD1+SSD2), без LVM 5](#_Toc210587939)

[Аппаратный сбой 2. Система на втором RAID1 (HDD5+HDD6), без LVM 6](#_Toc210587940)

[Аппаратный сбой 3. Система на RAID0 (SSD1+SSD2) + RAID1 (HDD5+HDD6) через LVM 6](#_Toc210587941)

[Аппаратный сбой 4. Система на RAID0 (SSD3+SSD4) — «жёлтый массив» 7](#_Toc210587942)

[Аппаратный сбой 5. Система на RAID1 (HDD) внутри LVM 7](#_Toc210587943)

[Программные сбои 8](#_Toc210587944)

[Программный сбой 1. RAID (mdadm) не собирается автоматически (битые/потерянные метаданные) 8](#_Toc210587945)

[Программный сбой 2. Неправильное имя массива (md1 ожидался, а собрался md127) / несоответствие имени 8](#_Toc210587946)

[Программный сбой 3. LVM-метаданные повреждены / VG не найден 9](#_Toc210587947)

[Программный сбой 4. VG «partial»: LVM отказывается активировать root из-за «missing PV» 9](#_Toc210587948)

[Программный сбой 5. Initramfs не содержит поддержки mdraid/LVM или нужных конфигов 9](#_Toc210587949)

[Программный сбой 6. Несоответствие UUID (fstab/GRUB указывают старые идентификаторы) 10](#_Toc210587950)

[Программный сбой 7. Гонка инициализации (udev/порядок появления устройств, таймауты) 10](#_Toc210587951)

[Ответ 11](#_Toc210587952)

[Задача 2. Найди 10 отличий: Nginx/Apache 11](#_Toc210587953)

[Постановка задачи 11](#_Toc210587954)

[Решение 13](#_Toc210587955)

[Nginx.conf 13](#_Toc210587956)

[Все ошибки в nginx.conf 13](#_Toc210587957)

[Исправленный файл конфигурации nginx.conf 13](#_Toc210587958)

[Apache.conf 14](#_Toc210587959)

[Все ошибки в apache.conf 14](#_Toc210587960)

[Исправленный файл конфигурации apache.conf 15](#_Toc210587961)

[Задание 3. Задача 3. Shell-scripting vs «Однострочники» 15](#_Toc210587962)

[Постановка задачи 15](#_Toc210587963)

[Решение 16](#_Toc210587964)

# Задача 1. Шерлок Холмс и LVM+RAID

## Постановка задачи

Условие

На сервере 6 дисков (4 SSD, 2 HDD):

Диски SSD #1 и SSD #2 объединены в RAID 0.

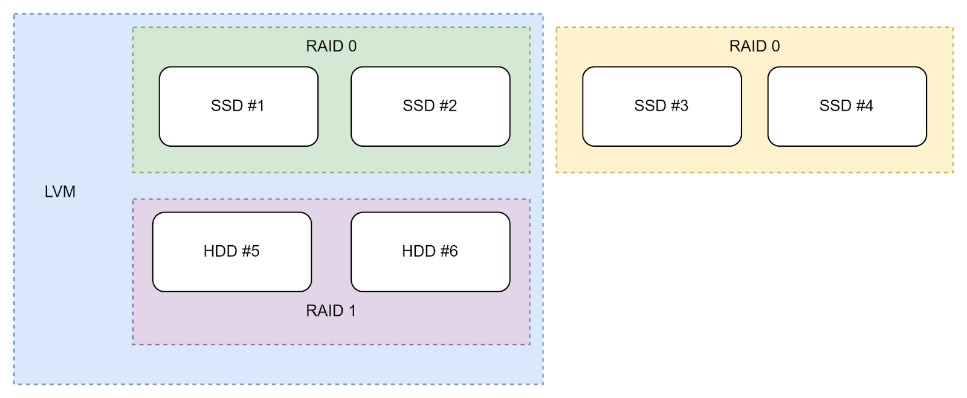
Диски SSD #3 и SSD #4 объединены в RAID 0.

Диски HDD #5 и HDD #6 объединены в RAID1.

Вопрос

Система на сервере перестала запускаться. Имея вводные из условия, с каким или какими дисками произошла проблема, если точно поломка в нем или них? Почему?

Ответ поясните. Вспомните об особенностях настройки LVM.

****

## Объяснение решения

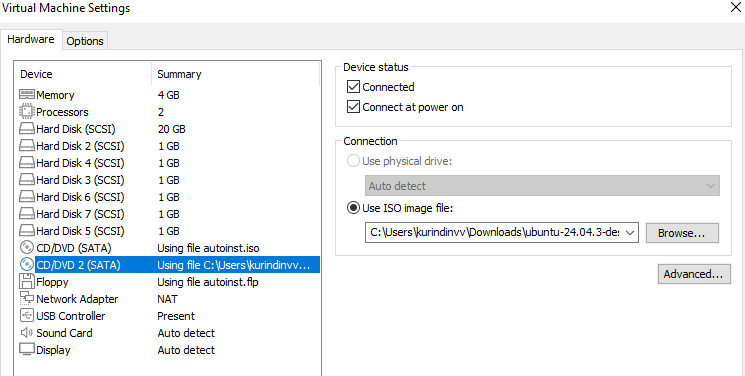
Так как в задаче четко не указано, что проблема в дисках, рассмотрим как аппаратные, так и проагрммный сбои. Фразу «Имея вводные из условия, с каким или какими дисками произошла проблема, если точно поломка в нем или них?» читаю так, что в чем поломка, в дисках или нет , не понятно. Фраза «Если точно поломка в нем или в них?» может быть трактована неоднозначно. Что значит: Если? Если надо четко дать задачу, то так и надо написать Предусловия: поломка в дисках. Тем не менее , прошу обратить внимание, что в ответе будут указаны и аппаратные сбои.

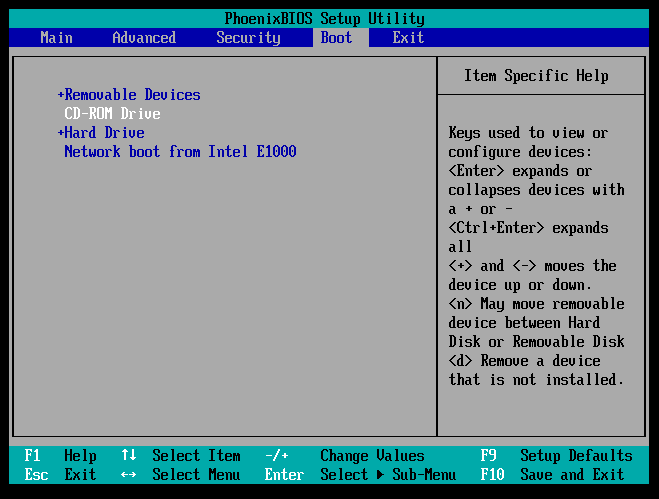
## Решение

### Настройка VMware для запуска с Live USB

Так как система не запускается, то нам надо загрузиться с live USB и попытаться начать выяснять , в чем реально у нас проблема.

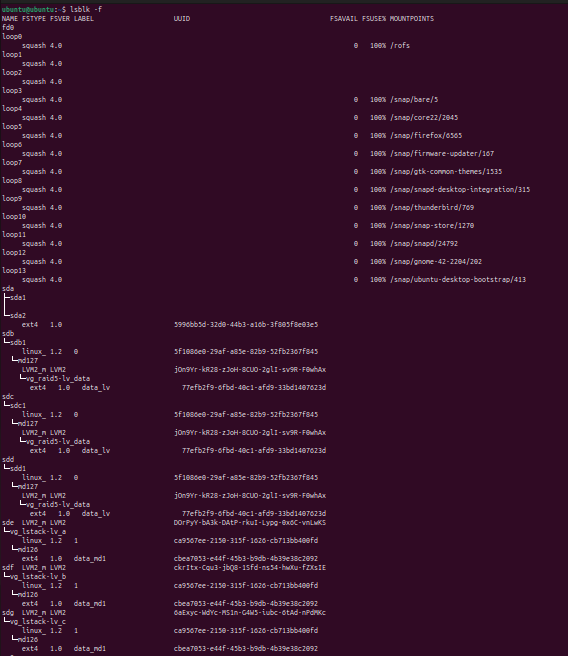
Для этого сначала подправим bios –загрузки.



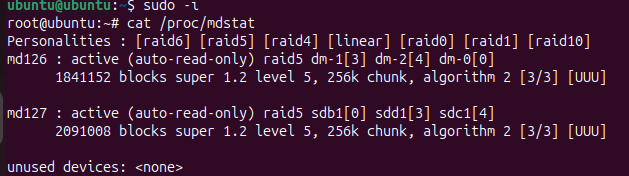


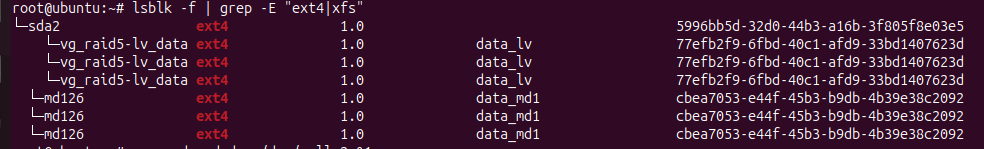
Потом надо не забыть постаивть CD-ROM Drive после hard drive.

Вот,что я получил, загрузившись с live usb на своей виртуалке с Ubuntu.



Далее видим результаты выполнения предыдущих домашних заданий





### Подготовка

sudo –i #сразу становимся под root тут.

# возможно потребуется установить утилиты (Ubuntu/Debian Live):

command -v mdadm >/dev/null 2>&1 || { apt-get update && apt-get install -y mdadm; }

command -v lvs >/dev/null 2>&1 || { apt-get update && apt-get install -y lvm2; }

mkdir -p /mnt/target

mdadm --assemble --scan --readonly || true

vgchange -ay --partial || true

lsblk -f

cat /proc/mdstat

Далее идут уже мои теоретические изыскания, потому что проверить на практике систему под рейд я смог из-за ограниченного количества времени.

### Аппаратные сбои

#### Аппаратный сбой 1. Система на первом **RAID0 (SSD1+SSD2), без LVM**

**Признаки в выводе:**

mount -o ro /dev/md0 /mnt/target || echo "Не монтируется /dev/md0"

# если /boot отдельный:

BOOT\_DEV=$(awk '$2=="/boot"{print $1;exit}' /mnt/target/etc/fstab); [ -n "$BOOT\_DEV" ] && mount -o ro "$BOOT\_DEV" /mnt/target/boot || true

awk '$2=="/"{print}' /mnt/target/etc/fstab

grep -E 'root=(UUID=|/dev/md|/dev/mapper/)' /mnt/target/boot/grub/grub.cfg | tail -n1

pvs; vgs; lvs -a -o +devices

umount /mnt/target/boot 2>/dev/null || true

umount /mnt/target 2>/dev/null || true

**Вывод:** система на массиве RAID0 (SSD1+SSD2). Очень рискованная схема: поломка любого одного SSD = падение ОС.

**Результат**: поломка диска тут может привести к неработоспособности системы

#### Аппаратный сбой 2. Система на втором **RAID1 (HDD5+HDD6), без LVM**

**Признаки в выводе:**

mount -o ro /dev/md1 /mnt/target || echo "Не монтируется /dev/md1"

BOOT\_DEV=$(awk '$2=="/boot"{print $1;exit}' /mnt/target/etc/fstab); [ -n "$BOOT\_DEV" ] && mount -o ro "$BOOT\_DEV" /mnt/target/boot || true

awk '$2=="/"{print}' /mnt/target/etc/fstab

grep -E 'root=(UUID=|/dev/md|/dev/mapper/)' /mnt/target/boot/grub/grub.cfg | tail -n1

pvs; vgs; lvs -a -o +devices

umount /mnt/target/boot 2>/dev/null || true

umount /mnt/target 2>/dev/null || true

**Вывод:** система на зеркале HDD (RAID1).

**Результат**: поломка диска тут ни на что не повлияет, если не вылетят оба диска

#### Аппаратный сбой 3. Система на **RAID0 (SSD1+SSD2) + RAID1 (HDD5+HDD6) через LVM**

**Признаки в выводе:**

lvs -a -o lv\_path,lv\_name,vg\_name,devices --noheadings

ROOT\_LV=$(lvs --noheadings -o lv\_path,lv\_name | awk 'tolower($2) ~ /(^|[-\_])root$/{print $1; exit}')

mount -o ro "$ROOT\_LV" /mnt/target

BOOT\_DEV=$(awk '$2=="/boot"{print $1;exit}' /mnt/target/etc/fstab); [ -n "$BOOT\_DEV" ] && mount -o ro "$BOOT\_DEV" /mnt/target/boot || true

awk '$2=="/"{print}' /mnt/target/etc/fstab

grep -E 'root=(UUID=|/dev/md|/dev/mapper/)' /mnt/target/boot/grub/grub.cfg | tail -n1

lvs -a -o lv\_name,devices | grep -E '(root|lv\_root)'

umount /mnt/target/boot 2>/dev/null || true

umount /mnt/target 2>/dev/null || true

**Вывод:** система размазана по обоим массивам. Сбой одного SSD из RAID0 ломает всё.

**Результат**: поломка диска тут может привести к неработоспособности системы

#### Аппаратный сбой 4. Система на **RAID0 (SSD3+SSD4) — «жёлтый массив»**

**Признаки в выводе:**

mount -o ro /dev/md2 /mnt/target || echo "Не монтируется /dev/md2"

BOOT\_DEV=$(awk '$2=="/boot"{print $1;exit}' /mnt/target/etc/fstab); [ -n "$BOOT\_DEV" ] && mount -o ro "$BOOT\_DEV" /mnt/target/boot || true

awk '$2=="/"{print}' /mnt/target/etc/fstab

grep -E 'root=(UUID=|/dev/md|/dev/mapper/)' /mnt/target/boot/grub/grub.cfg | tail -n1

pvs; vgs; lvs -a -o +devices

umount /mnt/target/boot 2>/dev/null || true

umount /mnt/target 2>/dev/null || true

**Вывод:** система на RAID0 из SSD3+SSD4. Тоже крайне ненадёжно: один диск — и ОС падает.

**Результат**: поломка диска тут может привести к неработоспособности системы

#### Аппаратный сбой 5. Система на RAID1 (HDD) внутри LVM

**Признаки в выводе:**

lvs -a -o lv\_path,lv\_name,vg\_name,devices --noheadings

lvs -a -o lv\_name,devices | grep -E '(root|lv\_root)'

ROOT\_LV=$(lvs --noheadings -o lv\_path,lv\_name | awk 'tolower($2) ~ /(^|[-\_])root$/{print $1; exit}')

mount -o ro "$ROOT\_LV" /mnt/target

BOOT\_DEV=$(awk '$2=="/boot"{print $1;exit}' /mnt/target/etc/fstab); [ -n "$BOOT\_DEV" ] && mount -o ro "$BOOT\_DEV" /mnt/target/boot || true

awk '$2=="/"{print}' /mnt/target/etc/fstab

grep -E 'root=(UUID=|/dev/md|/dev/mapper/)' /mnt/target/boot/grub/grub.cfg | tail -n1

umount /mnt/target/boot 2>/dev/null || true

umount /mnt/target 2>/dev/null || true

**Вывод:** формально система в LVM, но фактически она сидит только на RAID1. В этом случае отказ SSD не ломает загрузку.

**Результат**: поломка одного диска тут ни на что не повлияет

На текущий момент имеем, что проблемы с невозможностью загрузить систему мы имеем при:

* Аппаратном сбое 1 «Система на первом **RAID0 (SSD1+SSD2)», без LVM ,**
* Аппаратном сбое 3 «Система на **RAID0 (SSD1+SSD2) + RAID1 (HDD5+HDD6) через LVM»,**
* Аппаратном сбое **4 «Система на RAID0 (SSD3+SSD4) — «жёлтый массив»**

Далее смотрим программные проблемы

### Программные сбои

#### Программный сбой 1. RAID (mdadm) не собирается автоматически (битые/потерянные метаданные)

**Признаки в выводе:**

mdadm --detail --scan

cat /proc/mdstat

lsblk -f | grep -E 'md[0-9]|sd[a-z]'

**Вывод:** диски видны, но массив не собран из-за метаданных/конфига mdadm.

**Результат:** система не загружается, даже при «живых» дисках (root на недоступном md-устройстве).

#### Программный сбой 2. Неправильное имя массива (md1 ожидался, а собрался md127) / несоответствие имени

**Признаки в выводе:**

# что ожидает установленная система:

grep -E 'root=(UUID=|/dev/md|/dev/mapper/)' /mnt/target/boot/grub/grub.cfg

# что реально собрано в live:

cat /proc/mdstat

mdadm --detail –-scan

**Вывод:** несоответствие между тем, что ожидает загрузчик/fstab, и тем, как ядро/udev назвали массив.

**Результат:** система не находит root-устройство и уходит в emergency shell.

#### Программный сбой 3. LVM-метаданные повреждены / VG не найден

**Признаки в выводе:**

cat /proc/mdstat

pvs; vgs; lvs -a -o +devices

**Вывод:** RAID в порядке, но LVM-слой сломан (метаданные VG/LV повреждены/утеряны).

**Результат:** система не загружается, т.к. root-LV недоступен при целых массивах.

#### Программный сбой 4. VG «partial»: LVM отказывается активировать root из-за «missing PV»

**Признаки в выводе:**

vgs

lvs -a -o +devices | grep '(missing)'

vgchange -ay --partial

**Вывод:** LVM считает группу неполной (ссылается на отсутствующий PV/UUID), даже если RAIDы собраны — несогласованность метаданных VG и фактических устройств.

**Результат:** root не активируется; загрузка невозможна без вмешательства.

#### Программный сбой 5. Initramfs не содержит поддержки mdraid/LVM или нужных конфигов

**Признаки в выводе:**

KVER=$(ls /mnt/target/boot/initrd.img-\* 2>/dev/null | sed 's|.\*/initrd.img-||' | tail -n1)

[ -n "$KVER" ] && lsinitramfs "/mnt/target/boot/initrd.img-$KVER" | \

grep -E '(^etc/mdadm|/lvm|/dm-|^usr/lib/modules/.\*/kernel/drivers/md/)'

**Вывод:** образ initramfs собран без модулей/конфигов для mdraid/LVM, поэтому на стадии initrd корневое устройство недоступно.

**Результат:** система не находит root на ранней фазе и падает в initramfs shell.

#### Программный сбой 6. Несоответствие UUID (fstab/GRUB указывают старые идентификаторы)

**Признаки в выводе:**

# UUID из установленной ОС:

awk '$2=="/"{print $1}' /mnt/target/etc/fstab

# Сопоставляем с реальностью в live:

blkid

# Для md/LVM дополнительно:

mdadm --detail --scan

pvs -o+pv\_uuid; vgs -o+vg\_uuid; lvs -a -o+lv\_uuid

**Вывод:** после пересборки/миграции изменились UUID массива/раздела/VG, а конфигурационные файлы остались старые.

**Результат:** root не монтируется из-за неверных ссылок по UUID.

#### Программный сбой 7. Гонка инициализации (udev/порядок появления устройств, таймауты)

**Признаки в выводе:**

# если журналы persistent:

journalctl --directory=/mnt/target/var/log/journal -b -p err | \

grep -E 'Timed out waiting for device|A start job is running'

# иначе смотрим dmesg старта, если сохранён:

grep -i 'Timed out waiting for device' /mnt/target/var/log/\* 2>/dev/null

**Вывод:** устройства md/LVM появляются с запозданием относительно попытки монтирования root (порядок/тайминги systemd-udev).

**Результат:** загрузка может срываться/зависать, хотя аппартные ресурсы и метаданные в порядке.

## Ответ

Аппаратный сбой 1. Система на первом RAID0 (SSD1+SSD2), без LVM

* Отказ одного SSD приводит к полной недоступности массива.
* Система не загружается.
* При выходе из строя двух SSD результат аналогичен (массив полностью потерян).

Аппаратный сбой 2. Система на втором RAID1 (HDD5+HDD6), без LVM

* Отказ одного HDD не нарушает работоспособность массива.
* При отказе двух HDD массив становится недоступен, система не загружается.
* Возможный частный случай: если загрузчик установлен только на одном HDD, отказ именно этого диска делает систему не загружаемой.

Аппаратный сбой 3. Система на RAID0 (SSD1+SSD2) + RAID1 (HDD5+HDD6) через LVM

* Отказ одного SSD в RAID0 делает недоступной часть LVM, система не загружается.
* При одновременном отказе двух HDD массив RAID1 теряется, LVM становится частично недоступен, система не загружается.

Аппаратный сбой 4. Система на RAID0 (SSD3+SSD4) — «жёлтый массив»

* Отказ одного SSD приводит к полной недоступности массива.
* Если на массиве размещены системные разделы (/, /boot), система не загружается.
* Если массив использовался только для данных, отказ не влияет на загрузку.
* При отказе двух SSD результат аналогичен (массив полностью потерян).

Аппаратный сбой 5. Система на RAID1 (HDD5+HDD6) внутри LVM

* Отказ одного HDD не нарушает работоспособность массива.
* При отказе двух HDD массив становится недоступен, система не загружается.
* При установке загрузчика только на один HDD возможна недоступность системы при отказе именно этого диска.

И учитываем , что при любом из перечисленных 7-ми программных сбоев, система также не загрузится.

# Задача 2. Найди 10 отличий: Nginx/Apache

## Постановка задачи

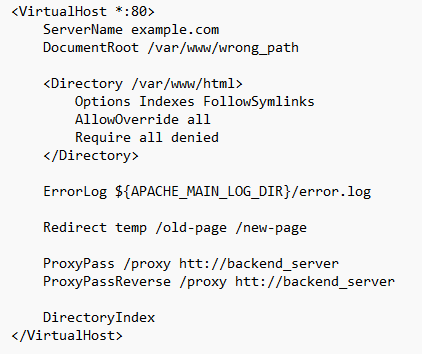
Условие

Даны два конфигурационных файла настройки веб-серверов Nginx и Apache. Необходимо найти 10 ошибок и написать исправленные конфигурационные файлы в поле ответа.

***nginx.conf***

****

***apache.conf***

****

## Решение

### Nginx.conf

#### Все ошибки в nginx.conf

1. index index.htm;  
   Указан только index.htm, а стандартный index.html не упомянут → при наличии index.html будет 403/404.  
   Исправление: index index.html index.htm;
2. return 302 /new-page;  
   Используется относительный путь, т.е редирект зависит от контекста. Лучше указывать абсолютный URL.  
   Исправление: return 302 http://example.com/new-page;
3. proxy\_pass http://backend\_server;  
   Недопустимое имя хоста. Подчёркивание в DNS нельзя.

+ тут пропущена буква p в http.   
Исправление: proxy\_pass http://backend-server; или IP.

1. **location /proxy { proxy\_pass http://backend-server; }**  
   Ошибка в склейке путей: при отсутствии хвостового / к апстриму уйдёт /proxy/test, а не /test.  
   Исправление:

location /proxy/ {

proxy\_pass http://backend-server/;

}

1. **Нет заголовков для бэкенда (**proxy\_set\_header**)**  
   Без Host, X-Real-IP, X-Forwarded-For, X-Forwarded-Proto бэкенд может работать некорректно.  
   Исправление: добавить proxy\_set\_header …

#### Исправленный файл конфигурации nginx.conf

server {

listen 80;

server\_name example.com;

root /var/www/html;

location / {

index index.html index.htm;

}

location /old-page {

return 302 http://example.com/new-page;

}

location /proxy/ {

proxy\_pass http://backend-server/;

# Рекомендуемые заголовки для корректной работы бэкенда

proxy\_set\_header Host $host;

proxy\_set\_header X-Real-IP $remote\_addr;

proxy\_set\_header X-Forwarded-For $proxy\_add\_x\_forwarded\_for;

proxy\_set\_header X-Forwarded-Proto $scheme;

}

}

### Apache.conf

#### Все ошибки в apache.conf

1. **DocumentRoot /var/www/wrong\_path**  
   Несуществующий путь, а в <Directory> указан /var/www/html. Несогласованность.  
   Исправление: DocumentRoot /var/www/html
2. **Require all denied**  
   Полный запрет доступа → сайт отдаст 403.  
   Исправление: Require all granted
3. **DirectoryIndex пустой**  
   Директива без значения вызовет ошибку.  
    Исправление: DirectoryIndex index.html
4. **ErrorLog ${APACHE\_MAIN\_LOG\_DIR}/error.log**  
   Такой переменной нет. В Ubuntu/Debian используют ${APACHE\_LOG\_DIR}. Исправление: ${APACHE\_LOG\_DIR}/error.log
5. **Options Indexes FollowSymLinks**  
   Indexes включает листинг каталогов — небезопасно.  
   Исправление: Options FollowSymLinks
6. **ProxyPass /proxy http://backend\_server**
   * Неверное имя хоста (подчёркивание).
   * Нет хвостового / → URLы будут склеиваться неправильно.  
     Исправление:

ProxyPass /proxy/ http://backend-server/

ProxyPassReverse /proxy/ http://backend-server/

1. **Redirect temp /old-page /new-page**  
   Здесь редирект тоже относительный → лучше указывать абсолютный URL.  
   Исправление: Redirect temp /old-page <http://example.com/new-page>
2. Дополнительно отмечу, что названия apache2.conf нет , есть , как у меня, например (Fedora), /etc/httpd/conf/httpd.conf или Debian/Ubuntu — /etc/apache2/apache2.conf
3. Также есть опечатки в htt://backendserver целых две

#### Исправленный файл конфигурации apache.conf

<VirtualHost \*:80>

ServerName example.com

DocumentRoot /var/www/html

<Directory /var/www/html>

Options FollowSymLinks

AllowOverride All

Require all granted

</Directory>

ErrorLog ${APACHE\_LOG\_DIR}/error.log

Redirect temp /old-page http://example.com/new-page

ProxyPass /proxy/ http://backend-server/

ProxyPassReverse /proxy/ http://backend-server/

DirectoryIndex index.html

</VirtualHost>

# Задание 3. Задача 3. Shell-scripting vs «Однострочники»

## Постановка задачи

Написать shell-скрипт (однострочник), который проверяет права доступа на SSH-ключи, хранящиеся в директории ~/.ssh/. Не забудьте, что на системе может быть несколько пользователей, а также root. Скрипт должен проверить, что приватные ключи имеют права доступа 600, а публичные — 644. Если права на файл отличаются от ожидаемых, информация об этом должна быть записана в лог-файл ssh\_permission\_errors.log в формате “<название файла> <права доступа>”.

Пример лог файла:

/home/user/.ssh/id\_rsa 644

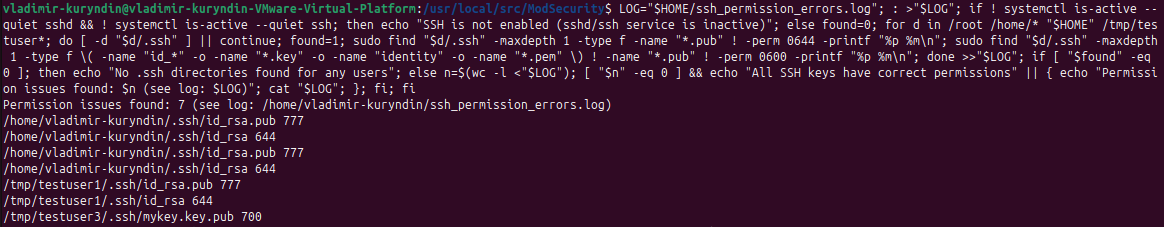
/home/user/.ssh/id\_rsa.pub 777

## Решение

Так как у меня никаких этих ключей на машине нет, создаим тестовые ключи и специально сделаем неправильные права.



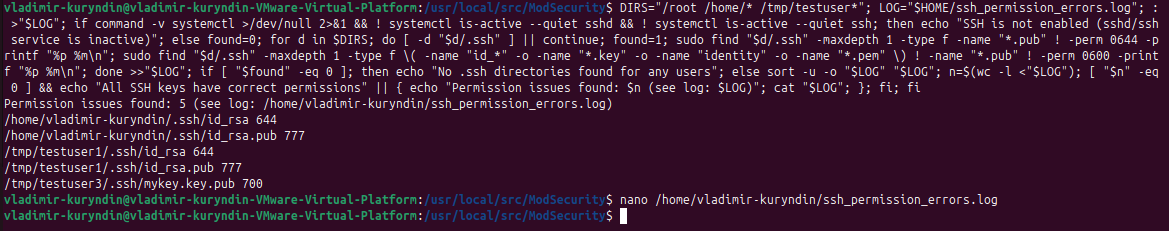
Добавим левых пользователей



Собственно, финальный скрипт.

DIRS="/root /home/\* /tmp/testuser\*"; LOG="$HOME/ssh\_permission\_errors.log"; : >"$LOG"; if command -v systemctl >/dev/null 2>&1 && ! systemctl is-active --quiet sshd && ! systemctl is-active --quiet ssh; then echo "SSH is not enabled (sshd/ssh service is inactive)"; else found=0; for d in $DIRS; do [ -d "$d/.ssh" ] || continue; found=1; sudo find "$d/.ssh" -maxdepth 1 -type f -name "\*.pub" ! -perm 0644 -printf "%p %m\n"; sudo find "$d/.ssh" -maxdepth 1 -type f \( -name "id\_\*" -o -name "\*.key" -o -name "identity" -o -name "\*.pem" \) ! -name "\*.pub" ! -perm 0600 -printf "%p %m\n"; done >>"$LOG"; if [ "$found" -eq 0 ]; then echo "No .ssh directories found for any users"; else sort -u -o "$LOG" "$LOG"; n=$(wc -l <"$LOG"); [ "$n" -eq 0 ] && echo "All SSH keys have correct permissions" || { echo "Permission issues found: $n (see log: $LOG)"; cat "$LOG"; }; fi; fi

Проверка на Ubuntu, где создавались тестовые пользователи и фейковые ключи.



Проверка на Fedora, где не было экспериментов с ssh

