Учебный центр АйТи Клауд

СОВРЕМЕННЫЕ ТЕХНОЛОГИИ ПРОСТЫМ ЯЗЫКОМ

АйТи Клауд –

один из ведущих Учебных центров России в сфере ИТ-образования с 1996 года

Преподаватель

Груздев Павел

Темы и направления деятельности:
- Администрирование Linux систем
- Внедрение и эксплуатация Zabbix

Учебный центр АйТи Клауд

Административное управление GNU/Linux

современные технологии простым языком

Административное управление GNU/Linux.
План курса

- Глава 1. Написание сценариев Bash.
- Глава 2. Процесс загрузки и уровни выполнения.
- Глава 3. Управление устройствами.
- Глава 4. Настройка сетевых параметров.
- Глава 5. Настройка удаленного доступа.
- Глава 6. Печать в GNU/Linux.
- Глава 7. Системные журналы.
- Глава 8. Мониторинг системы.
- Глава 9. Отложенное и регулярное выполнение заданий.
- Глава 10. Резервное копирование.
- Глава 11. Программные системы хранения.

Учебный центр АйТи Клауд

Модуль 1. Написание сценариев Bash

современные технологии простым языком

Модуль 1. Написание сценариев Bash. План

- Сценарии оболочки.
- Использование переменных оболочки.
- Интерактивная установка значений переменных.
- Позиционные параметры.
- Оператор test.
- Условное исполнение команд.
- Оператор case.
- Циклы.
- Функции.

Сценарии оболочки

- Сценарий – текстовый файл с командами и синтаксическими конструкциями
- Команды выполняются последовательно
- Могут запускаться как команды или как аргумент при запуске оболочки

Сценарии оболочки

- Для отладки полезны опции -v или –x
- Конструкция #! – указывает оболочку, которая исполняет команды (shebang)
- Команда source или . – inline подстановка

Задание

- Создание сценария
- Запуск сценария
- Шебанг

Использование переменных оболочки

- Переменные – строки
- Можно выполнять простейшие арифметические действия
- Имена чувствительны к регистру
- Имя должно начинаться с буквы или символа _

Использование переменных оболочки

- ${}
- ${имя:-значение}
- ${имя:=значение}
- ${имя:+значение}

Массивы

- declare
  - -A
  - -a
- unset

Интерактивная установка значений переменных

- Команда read – считывает значения переменных из стандартного потока ввода

Позиционные параметры

- Позиционные параметры – аргументы скрипта
- `$0` – имя команды;
- `$1` до `$9` – аргументы с 1 по 9
- `${10}…` – аргументы с 10 и далее
- Команда `shift` – сдвигает позиционные параметры

Позиционные параметры

- $* - строка из значений всех аргументов командной строки, разделенных символом разделителем по умолчанию, заданному в переменной IFS;
- $@ - строка из значений всех аргументов командной строки, разделенных пробелами;
- $# - количество аргументов командной строки;
- $? - код возврата предыдущей команды;
- $$ - PID оболочки.

Задание

- Использование переменных
- Считывание потока ввода
- Отладка
- Сдвиг и установка новых аргументов запуска

Оператор test

- Проверка существования и атрибутов файлов
- Сравнение файлов
- Проверка установки опций оболочки
- Сравнение строк
- Сравнение целых чисел
- Конструкция [ ] – синоним test

Условное исполнение команд

- `&&` - успешный запуск предыдущей команды
- `||` - неудачный запуск предыдущей команды
- `if условие`

```
  then
     команды
  fi
```

Оператор case

```
case слово in
  шаблон1 )
      команды
      ;;
  шаблон2 )
      команды
      ;;
  *)
      команды
      ;;
esac
```

Задание

- Проверка файлов и переменных
- Использование условных операторов

Циклы

- Цикл for для работы со списком

```
for переменная in список
do
      команда
done
```

- Цикл for со счетчиком

```
for ((i=1; i<=N ; i++))
do
      команда
done
```

Циклы

- Цикл while работает с предпроверкой условия

```
counter=0
while [[ $counter -le N ]]
do
      echo $counter
      (( counter++ ))
done
```

- Цикл until работает аналогично while

Задание

- Использование циклов

Функции

```
function имя()
{
   команды
}
```

Задание

- Использование функций

Что изучили в данном модуле

- Сценарии оболочки.
- Использование переменных оболочки.
- Интерактивная установка значений переменных.
- Позиционные параметры.
- Оператор test.
- Условное исполнение команд.
- Оператор case.
- Циклы.
- Функции.

Учебный центр АйТи Клауд

Модуль 2. Процесс загрузки и уровни выполнения

современные технологии простым языком

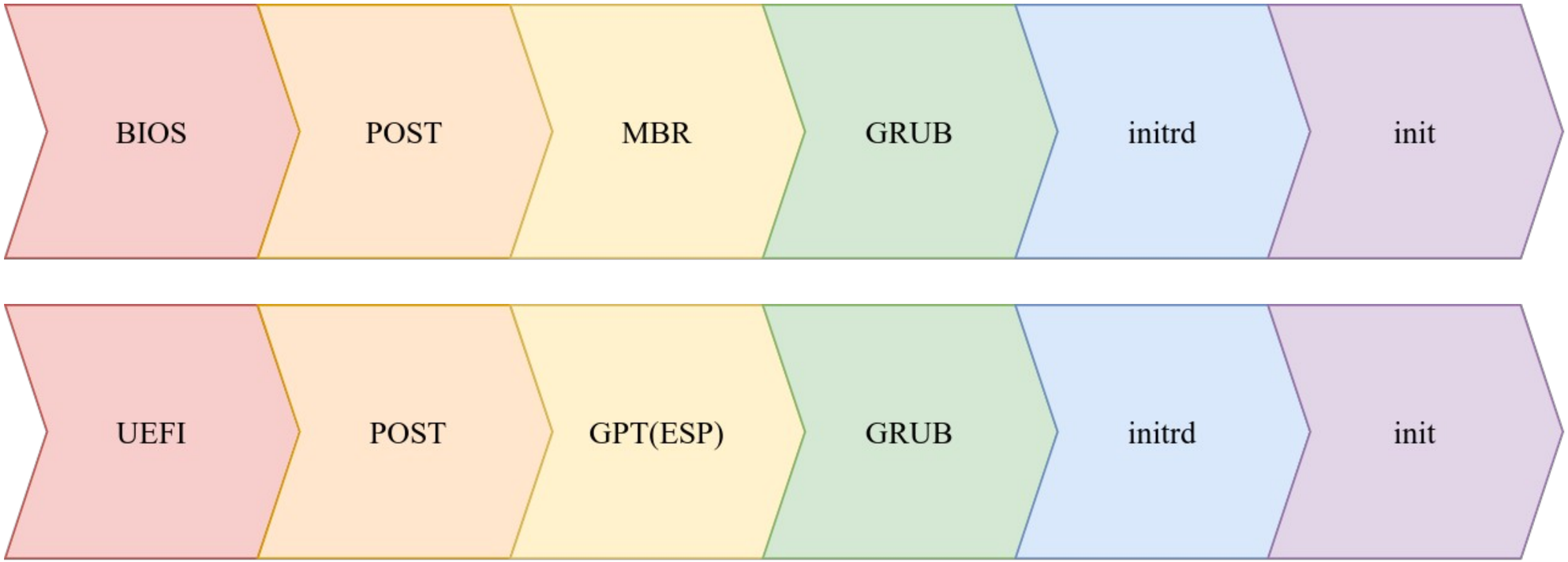
Модуль 2. Процесс загрузки и уровни выполнения. План

- Последовательность процесса загрузки
- Загрузчик GRUB2
- Инициализация системы с помощью демона systemd
- Остановка и перезагрузка системы

Последовательность процесса загрузки

- Процесс загрузки компьютера состоит из нескольких этапов
  - BIOS (UEFI)
    - Включение питания
    - POST
    - Bootloader
  - Загрузчик
    - Первая стадия (MBR)
    - Вторая стадия (основной загрузчик)
    - Поиск ядра и образа начальной загрузки
  - Запуск ядра
    - Запуск системных процессов
    - Инициализация системы демоном systemd (initd)

Последовательность процесса загрузки



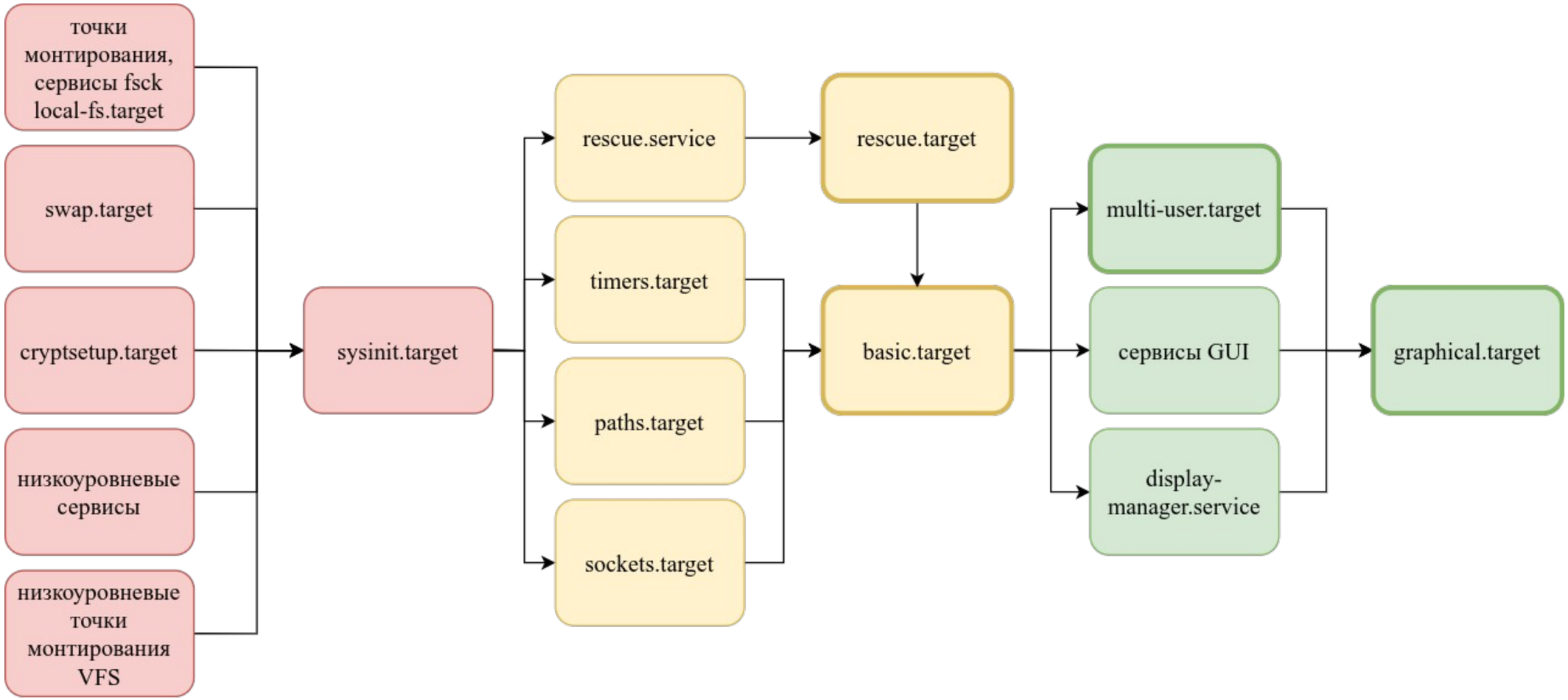
Загрузчик GRUB

- Grand Unified Bootloader — универсальный загрузчик Unix подобных систем (Linux, FreeBSD, Solaris ...)
- Поддерживает цепную загрузку (chainload)
- Имеется встроенная оболочка
- Первая версия почти не используется

Загрузчик GRUB2

- Основное новшество — возможность описать загрузку ОС без привязки к физическому расположению устройств
- Конфигурационный файл — `/boot/grub2/grub.cfg`
- В GRUB2 конфигурационный файл не редактируется вручную
- Вместо этого имеется специальная команда `grub2-mkconfig`
- Для смены параметров загрузки ядер Linux необходимо вносить изменения в файл `/etc/default/grub`

Инициализация системы с помощью демона systemd



Различные режимы загрузки

- В systemd отказались от понятия уровень исполнения вместо него используется понятие цель (target)
- Цель это некоторое состояние в которое должна попасть ОС.
- Одни цели могут зависеть от других.
- Цели так же зависят от других юнитов (служб, сокетов, монтирований и т.д.).

Зависимости целей

```
systemctl list-dependencies default.target | grep target
default.target
● └─multi-user.target
●   ├─basic.target
●   │ ├─paths.target
●   │ ├─slices.target
●   │ ├─sockets.target
●   │ ├─sysinit.target
●   │ │ ├─cryptsetup.target
●   │ │ ├─local-fs.target
●   │ │ └─swap.target
●   │ └─timers.target
●   ├─getty.target
●   ├─nfs-client.target
●   │ └─remote-fs-pre.target
●   └─remote-fs.target
●     └─nfs-client.target
●       └─remote-fs-pre.target
```

Unit

- Основополагающее понятие systemd — unit
- Unit — отдельный элемент для описания и управления в systemd
- С каждым юнитом может быть связан конфигурационный файл, который должен находиться в одном из каталогов:
  1. `/etc/systemd/system`
  2. `/run/systemd/system`
  3. `/usr/lib/systemd/system`

systemctl

- Команда `systemctl` управляет работой демона systemd

Остановка и перезагрузка системы

- Команды для остановки и перезагрузки
  - `systemctl` – основная команда управления питанием, остальные являются ссылками
  - `init`
  - `halt poweroff reboot`
  - `shutdown`

```
root@debian12:~# ls -l /usr/sbin/init
lrwxrwxrwx 1 root root 20 сен 20  2023 /usr/sbin/init -> /lib/systemd/systemd
root@debian12:~# ls -l /usr/sbin/shutdown
lrwxrwxrwx 1 root root 14 сен 20  2023 /usr/sbin/shutdown -> /bin/systemctl
root@debian12:~# ls -l /usr/sbin/halt
lrwxrwxrwx 1 root root 14 сен 20  2023 /usr/sbin/halt -> /bin/systemctl
```

Задание

- Управление загрузкой
- Управление инициализацией systemd

Что изучили в данном модуле

- Последовательность процесса загрузки
- Загрузчик GRUB2
- Инициализация системы с помощью демона systemd
- Остановка и перезагрузка системы

Учебный центр АйТи Клауд

Модуль 3. Управление устройствами

современные технологии простым языком

Модуль 3. Управление устройствами. План

- Установка нового оборудования
- Работа с модулями ядра
- UDEV и каталоги `/dev` и `/sys`
- Работа с устройствами.

Установка нового оборудования

- Подключение оборудования может быть
- произведено двумя основными способами:
  - в слоты расширения
  - к внешним шинам или портам
- Список прерываний - `/proc/interrupts`
- Адреса ввода-вывода - `/proc/ioports`
- Каналы DMA - `/proc/dma`

Работа с модулями ядра

- Модули ядра представляют собой объектные файлы с кодом, который можно подключать к работающему ядру Linux
- Модули располагаются в каталоге `/lib/modules/имя_ядра`
- Команда `uname -r` позволяет узнать имя ядра, запущенного в данный момент

Работа с модулями ядра

- Команды:
  - `lsmod insmod rmmod`
  - `modinfo`
  - `modprobe`
- Файлы:
  - `/proc/modules` — загруженные модули
  - `/etc/modprobe.d` — параметры загрузки модулей
  - `/etc/modules-load.d` — загрузка модулей при старте системы

UDEV и каталоги /dev и /sys

- `/dev` — содержит файлы устройств
- `/sys` — структуры ядра в виде файлов для управления устройствами
- UDEV — работающая в пространстве пользователя система, с помощью которой системный администратор может создавать обработчики событий

Правила UDEV

- С помощью правил вы можете настроить настроить как реагировать системе на подключение или отключение устройств
  - «Заводские» правила находятся в каталоге `/lib/udev/rules.d/`
  - Пользовательские правила UDEV следует располагать в каталоге `/etc/udev/rules.d`
  - Файлы с правилами должны иметь названия с суффиксом `.rules`

Мониторинг устройств

- `udevadm` предназначена для мониторинга и управления UDEV
- Для получения списка устройств PCI и USB вы можете воспользоваться утилитами `lspci` и `lsusb`

Диски и другие накопители

- SCSI диски - `/dev/sda, sdb, …`
- SD/MMC - `/dev/mmcblk0, mmcblk1, …`
- Virtio - `/dev/vda, vdb, …`
- Контроллеры NVMe - `/dev/nvme0, nvme1, …`
- Накопители NVMe - `/dev/nvme0n1, nvme0n2, …`

SMART

- В составе пакета smartmontools имеются утилита `smartctl` и демон `smartd`
  - `smartctl` используется для управления и получения данных от SMART
  - `smartd` предназначена для мониторинга и выполнения действий в случае возникновения проблем

Сетевые интерфейсы

- Для сетевых интерфейсов не создаются файлы устройств
- PNIDN - Predictable Network Interface Device Names
  1)Встроенные сетевые устройства: enoX
  2)Устройства PCI Express с горячим подключением: ensX
  3)Имена по расположению оборудования: enpXsY
  4)Именование по MAC адресу: enx112233445566
  5)Классический способ именования: ethX

Поддержка USB

- Как правило поддержка контроллеров включена непосредственно в ядро, а не в виде модулей
- Работает через подсистему SCSI
- Интерфейс имеет несколько версий
  - 1.1 драйверы OHCI и UHCI
  - 2.0 драйвер EHCI
  - 3.0 драйвер XHCI
  - Wireless WHCI
  - Virtual VHCI
- Проверка поддержки в ядре — `grep HCI_HCD /boot/config-$(uname -r)`
- Сведения о подключенных устройствах — `lsusb`

Виртуальные устройства

- /dev/null
- /dev/zero
- /dev/random и /dev/urandom
- /dev/loop*
- /dev/tty* и /dev/pts/*

Задание

- Модули ядра
- Работа с устройствами

Что изучили в данном модуле

- Установка нового оборудования
- Работа с модулями ядра
- UDEV и каталоги `/dev` и `/sys`
- Работа с устройствами

Учебный центр АйТи Клауд

Модуль 4. Настройка сетевых параметров

современные технологии простым языком

Модуль 4. Настройка сетевых параметров. План

- Сетевые интерфейсы
- Методы настройки сети
- Использование ifupdown
- Использование NetworkManager
- Использование systemd-networkd

Сетевые интерфейсы

- Программный интерфейс между процессами в ОС и сетевым адаптером
- Для сетевых интерфейсов не создаются файлы устройств
- Создаются автоматически
- В каталоге `/sys/class/net` имеются ссылки на устройства

```
sa@debian12:~$ ls /sys/class/net
enp0s3  lo
```

PNIDN

- PNIDN (Predictable Network Interface Device Names) используются в системе udev systemd.
- Примеры имен устройств:
  - Встроенные сетевые устройства: enoX (X — номер устройства)
  - Устройства PCI Express с горячим подключением: ensX (X — номер слота)
  - Имена по расположению оборудования: enpXsY (X — номер слота PCI, Y — номер устройства)
  - Именование по MAC адресу: enx112233445566
  - Классический способ именования: ethX

Методы настройки сети

- Ручной — командами
- Сценарии настройки сети — `ifup/ifdown` + файл (файлы) конфигурации
- Служба для настройки сети (NetworkManager, conman, systemd-networkd)

Ручная настройка

- `ip` — `ip [опции] объект [команда [параметры]]`
- `ifconfig` — `ifconfig [имя_интерфейса] [адрес] [опции]`

Сценарии настройки сети

- `ifup` — включить интерфейс
- `ifdown` — выключить интерфейс
- Расположение постоянных параметров для сценариев
  - Red Hat `/etc/sysconfig/network-scripts/ifcfg-*`
  - Debian `/etc/network/interfaces`

NetworkManager

- Сервис NetworkManager — автоматизированная настройка сетевых интерфейсов
- Полезен на рабочих станциях
- В графических средах имеются удобные приложения для управления соединениями NetworkManager
- Для управления соединениями NetworkManager из командной строки удобно использовать утилиту `nmcli`

systemd-networkd

- Является частью systemd
- Основная команда управления - `networkctl`
- Автоматически создает интерфейсы при появлении устройств в системе, также имеется возможность создавать виртуальные устройства
- Конфигурационные файлы находятся в `/lib/systemd/network` и `/etc/systemd/network`
- Каталог временных файлов — `/run/systemd/network/`

Средства диагностики сетей TCP/IP в Linux

- ping
- traceroute
- ss(netstat)
- lsof
- netcat (nc), socat
- telnet
- Wireshark, tcpdump, tshark
- nmap

- Основные причины неисправности

1. Неисправность в работе сетевых кабелей и/или активного сетевого оборудования (сетевых адаптеров, коммутаторов, концентраторов, маршрутизаторов и т.п.).
2. Несоответствие или отсутствие драйвера сетевого адаптера.
3. Неверные настройки IP адресов узлов сети.
4. “Засорение” кэша ARP.
5. Неверным указанием IP адреса маршрутизатора по умолчанию.
6. Ошибочная установка IP адреса DNS сервера или неверные соответствия IP адресов узлов их именам в /etc/hosts.
7. Блокирующие настройки фильтра IP пакетов.

Диагностика разрешения имен

- Утилиты:
  - `host`
  - `dig`
  - `nslookup`
  - `whois`

Задание

- Простая настройка сети

Что изучили в данном модуле

- Сетевые интерфейсы
- Методы настройки сети
- Использование ifupdown
- Использование NetworkManager
- Использование systemd-networkd

Учебный центр АйТи Клауд

Модуль 5. Настройка удаленного доступа

современные технологии простым языком

Модуль 5. Настройка удаленного доступа. План

- SSH
- XRDP
- Remmina
- VNC

SSH

- SSH — сетевой протокол, используемый для удалённого управления операционными системами и передачи файлов.
- Аббревиатура расшифровывается как Secure Shell.
- Ключевая особенность заключается в том, что SSH шифрует трафик, делая подключения безопасными.
- Реализован пакетом openssh (как серверная, так и клиентская части) и, обычно, устанавливается по умолчанию.

Команда SSH

- «Ssh» — это команда безопасного соединения, используемая в сети.

```
ssh user69@200.200.200.1
```

- Системные администраторы обычно используют ssh для подключения к серверам или сетевым устройствам, которыми они управляют

Конфигурационный файл /etc/ssh/sshd_config

- Основные параметры:
  - **Port 22** — задаёт порт работы сервера
  - **ListenAddress 0.0.0.0** — с какого интерфейса принимать подключения
  - **PermitRootLogin yes** — запрет доступа пользователю root
  - **LoginGraceTime 2m** — если пользователь не смог аутентифицироваться за это время, сервер разрывает соединение
  - **PubkeyAuthentication yes** — разрешает аутентификацию по публичным ключам
  - **PasswordAuthentication yes** — разрешает аутентификацию по паролям
  - **GSSAPIAuthentication yes** — разрешает аутентификацию по GSSAPI
  - **X11Forwarding yes** — разрешает отправлять через SSH графическую информацию
  - **AllowUsers someuser** — разрешает подключаться только пользователю someuser
  - **AllowGroups ssh_group** — разрешает подключаться только пользователям из группы ssh_group
  - **LogLevel INFO** — задаёт уровень логгирования

Аутентификация по ключам в SSH

- Создаём ключ:

```
ssh-keygen
```

  - Подтверждаем расположение ключа `~/.ssh/id_rsa`
  - Вводим кодовое слово, если нужно, подтверждаем выбор.
  - Создался закрытый (`~/.ssh/id_rsa` – если указан пароль, то ключ будет зашифрован) и открытый ключи (`~/.ssh/id_rsa.pub`).
  - Копируем открытый ключ на сервер, к которому будем подключаться, в файл `~/.ssh/authorized_keys`. Можно использовать команду

```
ssh-copy-id
```

Команда SCP

- «Scp» используется для безопасной передачи файлов между разными хостами. Вы копируете свои файлы на другое устройство в сети или можете получить эти файлы также с них с помощью scp.
- Вы можете копировать как файлы, так и каталоги. Для каталогов вы должны использовать дополнительный параметр «`-r`».
- Копирование с локального хоста на удаленный хост:

```
scp $filename user@targethost:remote_path
```

Советы по безопасному использованию SSH

- Запретите на удалённый root-доступ.
- Запретите подключения с пустым паролем или отключите вход по паролю.
- Используйте нестандартные порты для SSH-сервера.
- Используйте длинные SSH-ключей (2048 бит и более).
- Ограничьте список IP-адресов, с которых разрешён доступ или запретите доступ с некоторых потенциально опасных адресов.
- Не используйте распространённые или широко известные имена логинов.
- Регулярно просматривайте сообщения об ошибках аутентификации.
- Установите систему обнаружения вторжений (IDS: fail2ban, sshguard).

XRDP

- XRDP — это открытая реализация Microsoft Remote Desktop Protocol (RDP) для Unix-подобных систем
- Архитектура:
  - Клиент RDP (Remmina, Windows Remote Desktop, FreeRDP, ...) подключается к порту 3389
  - XRDP сервер принимает соединение и аутентифицирует пользователя
  - Менеджер сессий создаёт виртуальный X11 дисплей
  - Рабочий стол (GNOME, MATE, KDE, ...) запускается в этой сессии
- Альтернативы XRDP:
  - Для GNOME – gnome-remote-desktop
  - Для KDE – krdp

Установка и настройка xrdp

- Установите пакеты xrdp и xorgxrdp
- Файлы для настроки:
  - RDP сервера - /etc/xrdp/xrdp.ini
  - Управление сеансами - /etc/xrdp/sesman.ini
  - Сценарий для запуска сеанса пользователя - /etc/xrdp/startwm.sh

Remmina

- Программа-клиент для удаленного подключения
- Поддерживаемые протоколы:
  - SSH
  - RDP
  - VNC
  - SPICE
  - X2Go
  - WWW
  - EXEC

VNC

- Протокол VNC (Virtual Network Computing) — это универсальный способ удаленного управления графическим рабочим столом. В отличие от RDP, он работает на уровне передачи «картинки» (пикселей), а не графических команд системы
- Особенности:
  - **Кроссплатформенность**: серверная и клиентская часть имеется почти под все современные ОС, не только Linux
  - **Порты**: по умолчанию используется TCP 5900 (для первого дисплея :0). Если на одной машине запущено несколько сессий, они занимают порты 5901, 5902 и так далее.
  - **Безопасность**: классический VNC не шифрует трафик (кроме пароля при входе). Для безопасной работы через интернет его обязательно нужно «заворачивать» в SSH-туннель или использовать VPN.
  - **Скорость**: VNC требователен к пропускной способности канала (рекомендуется от 1–2 Мбит/с для комфортной работы), так как передает именно графические данные, а не команды отрисовки, как RDP.

Популярные реализации протокола VNC

- RealVNC: оригинальная реализация от создателей протокола
- TightVNC / TigerVNC: популярные бесплатные версии с открытым кодом, оптимизированные для медленных каналов
- UltraVNC: версия для Windows с дополнительными функциями вроде передачи файлов и чата
- noVNC: позволяет подключаться к уделенной сессии через браузер

Установка и настрока TigerVNC (простой вариант)

1. Установите пакет tigervnc-standalone-server
2. Создайте пароль для VNC
3. Настройте запуск графического окружения
4. Запустите сервер VNC
5. Сделайте проброс портов с SSH на клиентской машине
6. Подключитесь клиентом

Запуск TigerVNC как службы

1. Настройте сопостовление пользователей и виртуального дисплея
2. Настройте службу для автоматического перезапуска сессии пользователя
3. Включите и запустите службу для каждого пользователя

Использование noVNC

- noVNC строго говоря не VNC сервер, это клиент VNC, который позволяет подключаться к локальным сессиям через веб-браузер
- noVNC это вебслужба, выполняющая роль прокси между клиентом и VNC сервером
- Похожие функции может выполнять Apache Guacamole

Подключение к локальной сессии

- Используйте пакеты x11vnc или wayvnc, для трансляции текущего сеанса в VNC
- Сначала создайте пароль

```
x11vnc -storepasswd
```

- Затем запустите VNC сервер

```
x11vnc -localhost -usepw
```

Задание

- Настройка удалённого доступа

Что изучили в данном модуле

- SSH
- XRDP
- Remmina
- VNC

Учебный центр АйТи Клауд

Модуль 6. Печать в GNU/Linux

современные технологии простым языком

Модуль 6. Печать в GNU/Linux. План

- Система печати CUPS
- Печать в CUPS
- Управление принтерами в CUPS
- Управление очередью печати в CUPS

Система печати CUPS

- Система печати CUPS имеет удобный веб интерфейс для настройки и управления печатью http://localhost:631
- Командой `cupsctl` можно включить или выключить удаленное управление или печать
- Основной файл конфигурации `/etc/cups/cupsd.conf`
- Описание принтеров в `/etc/cups/printers.conf`

Печать в CUPS

- Основная команда для печати в CUPS - это `lp`
- Поддерживается так же команда `lpr`

Управление принтерами в CUPS

- Удобный способ для добавления, удаления или изменения принтеров — веб интерфейс
- Команда `lpadmin` управляет принтерами через командную строку
- Команда `lpoptions` задает параметры принтеров

Управление очередью печати в CUPS

- `lpstat` — состояние очереди печати
- `cupsenable/cupsdisable` — включение или выключение принтера, но не очереди к нему
- `cupsaccept/cupsreject` — включение или выключение очереди
- `cancel.cups` (cancel) — отмена печати задания
- `lpmove` — перемещение задания на печать

Задание

- Управление принтерами

Что изучили в данном модуле

- Система печати CUPS
- Печать в CUPS
- Управление принтерами в CUPS
- Управление очередью печати в CUPS

Учебный центр АйТи Клауд

Модуль 7. Системные журналы

современные технологии простым языком

Модуль 7. Системные журналы. План

- Служба rsyslog
- Журналы systemd
- Служба ротации журналов

Служба rsyslog

- В современных дистрибутивах для системного журналирования обычно применяется `rsyslogd`
- Место хранения журналов обычно `/var/log`
- Помимо системных журналов приложения могут создавать свои журналы

Настройка rsyslogd

- Основной конфигурационный файл `/etc/rsyslod.conf`
- В файле определяете каналы поступления событий и каналы передачи
- Могут применяться шаблоны, например для автоматического формирования имени файла или формата сообщений
- Параметр facility определяет источник события
- Параметр severity уровень важности

Журналы systemd

- Файл `/etc/systemd/journald.conf` определяет параметры журналирования systemd
- Для анализа журналов используется команда `journalctl`
- Журналы systemd бинарные

Служба ротации журналов

- Программа `logrotate` используется для ротации журналов
- Обычно `logrotate` запускается по расписанию демоном cron
- Файла конфигурации `logrotate` - `/etc/logrotate.conf`
- В `/var/lib/logrotate/logrotate.status` содержится статус ротации журналов

Задание

- Управление журналированием

Что изучили в данном модуле

- Служба rsyslog
- Журналы systemd
- Служба ротации журналов

Учебный центр АйТи Клауд

Модуль 8. Мониторинг системы

современные технологии простым языком

Модуль 8. Мониторинг системы. План

- Процедура мониторинга
- Загрузка процессора
- Использование памяти
- Мониторинг дисков
- Мониторинг сети

Процедура мониторинга

- Определите способ мониторинга производительности:
  - Оперативный (интерактивный)
  - Исторический (статистический)
- Создайте базовую линию (baseline)
- Определите какие компоненты системы являются потенциально узким местом

Потенциально узкие места в производительности

- Основные компоненты любой системы:
  - Процессор
  - Память
  - Диск
  - Сетевой интерфейс

Общий оперативный мониторинг

- Утилиты `*top`
  - `top` — в реальном времени показывает общую загрузку системы и активность процессов
  - `htop` — по функционалу похожа на `top`, но более дружелюбна
  - `btop` — продвинутый интерактивный монитор ресурсов для терминала с очень стильным графическим интерфейсом
  - `atop` — системный монитор, со встроенной службой для записи данных о производительности
- Системный монитор в графике
- Файлы в каталогах `/proc` и `/sys`

Пакет sysstat

- Пакет sysstat содержит утилиты измерения производительности системы Linux
- Пакет sysstat состоит из следующих утилит:
  - `sar` — собирает и выдаёт информацию о системной активности
  - `iostat` — сообщает об использовании ЦП и статистику ввода/вывода дисков
  - `tapestat` — сообщает статистику ленточных накопителей
  - `mpstat` — сообщает глобальную и по-процессорную статистику
  - `pidstat` — сообщает статистику по Linux задачам (процессам)
  - `sadf` — отображает информацию, собранную sar, в различных форматах
  - `cifsiostat` — сообщает статистику ввода/вывода по файловым системам CIFS
- Пакет содержит службы для сбора статистики в файлы

Загрузка процессора

- Утилиты
  - `*top`
  - `iostat`
  - `sar`
- Нужно следить за общей загруженностью процессора и длиной очереди к процессору
- Важны не текущие значения, но наблюдения за некоторый период времени

Использование памяти

- Утилиты
  - `*top`
  - `free`
  - `vmstat`
  - `sar`
- Нужно следить за объемом свободной памяти и использованием подкачки

Мониторинг дисков

- Использование дискового пространства:
  - `df`
  - `lsblk`
  - `du`
- Нагрузка на диск (операций чтения и записи):
  - `iotop`
  - `iostat`
- Следите так же за размером очереди к диску

Мониторинг сети

- Статистика интерфейса:
  - `ip`
  - `ifconfig`
- Использование сети приложениями:
  - `iftop`
  - `nethogs`
- Важно не количество полученных или переданных байт или пакетов, но их изменение в единицу времени

Исторический мониторинг

- Сведения можно собирать локально, но это потенциальная проблема, если система перестала работать
- Другое преимущество централизованного мониторинга — получение общей картины и реакции на отклонения от нормы
- Средства централизованного мониторинга: zabbix, nagios, catci, ...

Аппаратная диагностика

- Многие серверные платформы имеют независимый от установленной ОС интерфейс управления и мониторинга аппаратной части
- Протокол IPMI
- Пакет lm-sensors

Ручной анализ событий

- Утилита `grep` основной инструмент для получения сведений из журналов в ручном режиме
- Стандартные сообщения syslog состоят из нескольких частей:

```
дата время узел процесс сообщение
```

- Формат сообщений из журналов приложений может быть совершенно другим

Автоматизированный анализ событий

- Автоматизированный анализ можно поделить на несколько типов:
  - Ожидание определенного события для оперативного реагирования
  - Получение общих отчетов о работе
  - Сбор статистики
  - Анализ данных для планирования дальнейшего развития

Задание

- Мониторинг системы

Что изучили в данном модуле

- Процедура мониторинга
- Загрузка процессора
- Использование памяти
- Мониторинг дисков
- Мониторинг сети

Учебный центр АйТи Клауд

Модуль 9. Отложенное и регулярное выполнение заданий

современные технологии простым языком

Модуль 9. Отложенное и регулярное выполнение заданий. План

- Отложенное выполнение заданий с помощью `at`
- Автоматизация выполнения регулярных рутинных задач

Отложенное выполнение заданий с помощью at

- Команда `at` выполняет команды в указанное время
- Время может быть абсолютное или относительное
- Команды запускаются с переменными пользователя, который запустил `at`
- Для контроля доступа к `at` используются файлы `/etc/at.allow` и `/etc/at.deny`
- Команда `batch` запускает команды, когда загрузка системы снизится

Автоматизация выполнения регулярных рутинных задач

- Для выполнения заданий по расписанию используется демон cron
- В systemd можно создать юнит типа timer для запуска периодических заданий
- Служба anacron может запускать пропущенные периодические задания, в том случае, когда машина была выключена
- В отличие от cron в anacron указывается не точное время заданий, а желаемая периодичность

Формат указания времени в crontab

m h dom mon dow

Где

- m — минуты (0-59)
- h — часы (0-23)
- dom — дни месяца (1-31)
- mon — меяцы (1-12)
- dow — дни недели (0-7), 0 и 7 — воскресенье

Специальные символы

- * - любое значение
- / - периодичность, например */5
- - - интервал, например 1-3
- , - список, например 1,6,12,18

Порядок работы с индивидуальными таблицами

- Текущая таблица пользователя сохраняется во временном файле для редактирования: `crontab -l > crontab.eg`
- Временный файл редактируется в текстовом редакторе: `vi crontab.eg`
- Отредактированная таблица заданий запоминается: `crontab crontab.eg`

Контроль доступа к cron

- В файле `/etc/cron.allow` кому можно использовать cron. Остальным нельзя
- Файл `/etc/cron.deny` используется если нет `/etc/cron.allow` и в нем перечисляется кому нельзя использовать cron
- В некоторых случаях используется специальная группа для контроля доступа

Специальные каталоги

- В каталоги `/etc/cron.hourly`, `/etc/cron.daily`, `/etc/cron.weekly` и `/etc/cron.monthly` можно поместить сценарии или программы, которые будут запускаться соответственно ежечасно, ежедневно, еженедельно или ежемесячно

Задание

- Отложенные задания
- Периодические задания

Что изучили в данном модуле

- Отложенное выполнение заданий с помощью at
- Автоматизация выполнения регулярных рутинных задач

Учебный центр АйТи Клауд

Модуль 10. Резервное копирование

современные технологии простым языком

Модуль 10. Резервное копирование. План

- Планирование резервного копирования
- Команда `dd`
- Утилиты для сжатия файлов
- Команда `tar`
- Команда `cpio`

Планирование резервного копирования

- Что копируем
- Куда копируем
- Как часто
- Частота ротации носителей
- Кто копирует
- Контроль со стороны человека
- В какие моменты времени
- Хранение носителей резервных копий;
- Порядок восстановления утраченных данных
- Время на копирование
- Время на восстановление

Команда dd

- Команда `dd` осуществляет блочное копирование из файла в файл
- Часто применяется для создания образов дисков
- Основные опции
  - `if` — источник
  - `of` — местоназначение
  - `bs` — размер блока
  - `count` — количество блоков

Утилиты для сжатия файлов

- Утилиты сжатия сжимают каждый файл в отдельности
- Разные утилиты имеют схожий синтаксис команд, но разные алгоритмы работы
- Чем лучше сжимает, тем больше процессорных ресурсов требуется для сжатия
- Основные утилиты:
- `gzip` — самая быстрая, но не сильное сжатие
- `bzip2` — сжимает лучше чем `gzip`, но чуть медленнее
- `xz` — сжимает сильно, но требуется больше ресурсов CPU

Команда tar

- Режим работы `tar` задается первой опцией:
- `c` — create - создать
- `x` — extract - извлечь
- `t` — test - протестировать
- `A` — concatenate — объединить архивы
- `u` — update - обновить
- `d` — diff - сравнить архив и файлы
- `r` — append - добавить к архиву
- Опция `f` указывает файл с архивом, без нее будет использована лента

Команда cpio

- Команда работает со стандартными потоками
- Часто для создания списка файлов, которые подлежат архивации используется команда `file`
- Три основных режима `cpio`
  - `-o` (copy-out) — в архив
  - `-i` (copy-in) — из архива
  - `-p` (pass-through) — копирование в каталог

Задание

- dd
- Сжатие файлов
- Резервное копирование

Что изучили в данном модуле

- Планирование резервного копирования
- Команда `dd`
- Утилиты для сжатия файлов
- Команда `tar`
- Команда `cpio`

Учебный центр АйТи Клауд

Модуль 11. Программные системы хранения

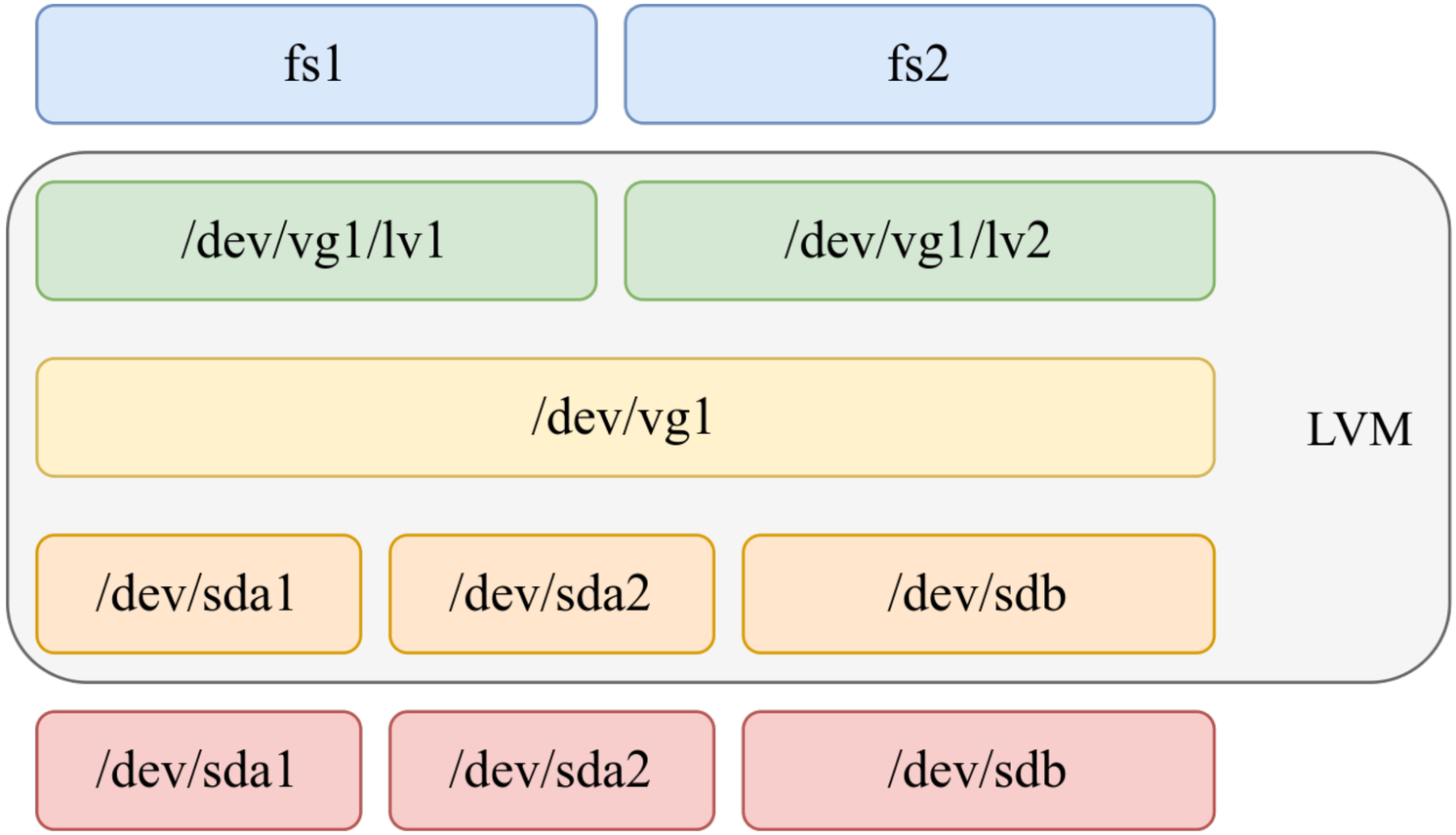
современные технологии простым языком

Модуль 11. Программные средства хранения. План

- Система LVM
- Программные RAID массивы

Система LVM

- Logical Volume Management — система обеспечивающая абстрактное управление устройствами хранения



Система LVM

- Последовательность действий
  1)Создать PV — `pvcreate`
  2)Создать VG — `vgcreate`
  3)Создать LV — `lvcreate`
- Получение информации о группах, томах — `vgdisplay`, `pvdisplay`, `lvdisplay`
- Управление группами и томами — `vg*`, `pv*`, `lv*`

RAID массивы

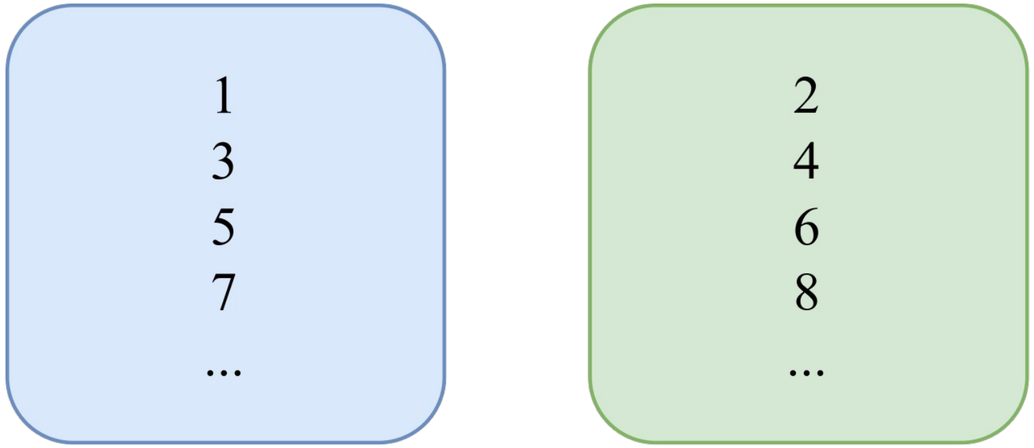
- Могут обеспечивать отказоустойчивость или производительность
- Уровни определяют способ чтения-записи информации в массиве

RAID массивы

- Линейный RAID (RAID -1)
  - Составной том — простое объединение нескольких дисков в общее пространство

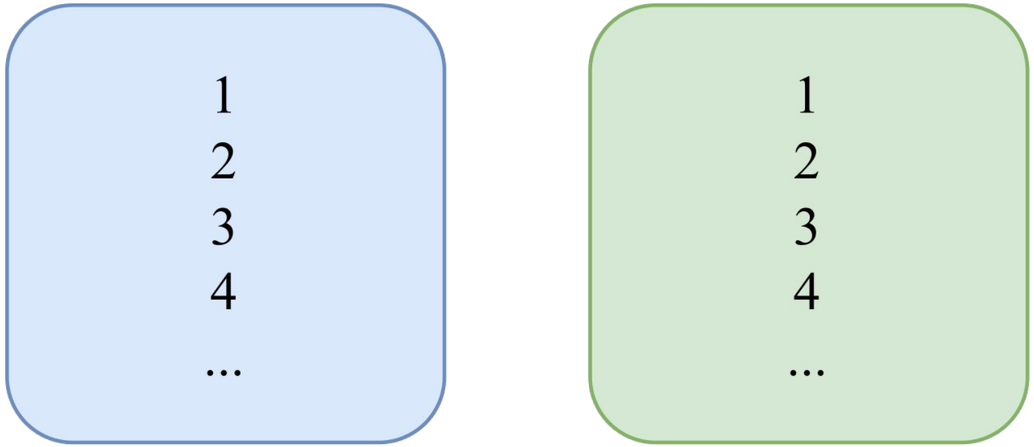
RAID массивы

- Чередующийся том (RAID0)
  - Чередование повышает скорость операций чтения-записи
  - Дисковое пространство разбивается на страйпы



RAID массивы

- Зеркальный том (RAID1)
  - Данные дублируются на дисках
  - Можно повысить скорость считывания данных



RAID массивы

- Чередующийся том с четностью (RAID5)
  - Повышенная отказоустойчивость при меньших затратах чем в зеркале

RAID массивы

- Управление программными массивами происходит через драйвер md
- Тип раздела для RAID следует устанавливать в `fd`
- Управление массивами осуществляется командой `mdadm`

Задание

- LVM
- Программный RAID

Что изучили в данном модуле

- Система LVM
- Программные RAID массивы

Спасибо, что выбрали обучение в УЦ АйТи Клауд!

Желаем успешно реализовать новые знания на практике!

И будем ждать новых встреч!

Связаться с преподавателем: ***pgruzdev@itcloud-edu.ru***

Наш сайт