

Task 1

1. С помощью vagrant развернуть 2 VM
2. Настроить репозиторий. Обновить систему. Установить Docker из репозитория
3. Установить **LVM**
 - создать LVG (Logical Volume Group) и добавить в нее существующие блочные устройства (минимум 2).
 - Установить Docker на основании официальной документации
 - запустить Docker container на базе произвольного Docker Image (образ выбираем в Docker Hub)
 - Настроить Docker device mapper на базе LVM (разделы device mapper используемые Docker должны иметь возможность автоматического расширения при достижении порогового значения уровня заполненности)
4. Установить систему применяя **RAID**
 - Создать 4 диска и подключить их к одной из VM
 - Из двух дисков создать RAID0 (фиксированная конфигурация)
 - Из двух дисков создать RAID1 (динамическая конфигурация)
 - Имитировать отказ одного из дисков массива
5. Проверить и настроить сеть (показать и объяснить корректность настроек)
6. Настроить sudo
7. Управление группами и пользователями
 - Создать группу
 - Создать пользователя и домашнюю директорию
 - Добавить пользователя в группу
 - Удалить ранее созданную группу, но пользователя и его группу оставить
 - Используя п. 6 обеспечить возможность простому пользователю перезапускать сервис (любой на выбор)
8. Установить и настроить ntp только на localhost и IP данной машины

Task 2

С помощью vagrant поднять 3 VM:

- Установить и настроить ssh (клиент и сервер)
- Сгенерировать RSA ключи размером 4K для текущего пользователя не заменяя существующие ключи
- Настроить использование ssh с учетом требований безопасности
- Залить ключ(и) на другую(-ие) машины
- Сделать доступ только по ключу и только для текущего пользователя, убрать root доступ, только IPv4 сеть, сменить стандартный порт у сервера и клиента у всех клиентских машин
- Настроить права доступа к файлам
- Сохранить с помощью scp настройки и конфиги в архив с помощью tar на хосте
- Создать 3 директории: current, new, old
- В директорию old файлы из любой существующей директории сохраняя изначальные атрибуты(права, владелец и т.д.)
- В директорию current скопировать файлы из любой существующей директории создавая атрибуты директории current
- В директорию new скопировать скопировать файлы из любой существующей директории с изменением атрибутов на атрибуты директории new (на 1 год назад)
- Создать tar архив из этих 3-х директорий
- Найти и сохранить список всех файлов в /bin/, которые написаны на Bash/Shell (в одну строку выполнение) в файл bin.txt
- Скачать iso файл (Debian net-inst) с интернета в текущую директорию, с учётом нестабильного соединения, используя wget
- Скачать репозиторий Git проекта (любого) используя curl

1. Network utilities & troubleshooting

- Сохранить текущий LA (Load Average) в файл la.txt
- Сохранить текущий загрузку дисков (размеры) в файл hdd.txt
- Сохранить текущий список процессов пользователей в файл pl.txt
- Сохранить информацию о SWAP-е в файл swap.txt
- Сохранить список всех SATA устройств в файл dev.txt
- Сохранить информацию у процессоре в cpu.txt

- Сохранить информацию о любом процессе в файл `<id_процесса>.txt`
- Сохранить информацию о партициях в файл `parts.txt`
- Проверить реальное соединение по порту (например 80 на любом сервере)
- Получить список всех открытых портов на удалённом сервере и сохранить в файл `ports.txt`
- Получить список всех TCP портов на машине и сохранить в файл `tcp.txt`
- Получить список всех UDP портов на машине и сохранить в файл `udp.txt`
- Получить список всех UNIX Socket портов на машине и сохранить в файл `usocket.txt`
- Получить таблицу маршрутизации на машине и сохранить в файл `net.txt`
- Получить статистику для каждого протокола на машине и сохранить в файл `nstat.txt`
- Изменить имя машины
- Получить список DNS и сохранить в файл `dns.txt`

Task 3

Создайте 3 виртуальные машины с:

Настройте виртуальные машины:

VM1 - должна иметь подключение к общедоступному Интернету через собственный интерфейс `nat`, VM2, VM3 должны быть доступны через `host-only adapter` имеющий `private IP`

VM2 - должна иметь доступ к Интернету через интерфейс `nat` VM1, VM3 и VM1 должны быть доступны через `host-only adapter` имеющий `private IP`

VM3 - должна быть доступна для VM1, VM2 через `host-only adapter` имеющий `private IP`, без доступа в Интернет

Конфигурационные файлы адаптеров VM с каждой виртуальной машины (6 файлов) и список правил `iptables / firewall-cmd` (1 файл) и команды, которые использовались для настройки `iptables / firewall-cmd` залить в гит