**LINUX 14.11**

Три джентельменских соглашения при использовании LINUX оп.

1. Текст всегда в двойных кавычках **“**Hello World**”**
2. Все переменные пишутся **СAPS LOCK** -ом.
3. У архивов всегда **двойное расширение** .tar.gz
4. Команда - **export** (пробел)НАЗВАНИЕРЕМЕНЕННОЙ=то, что ми ложем в переменную.
5. Команда - **env** - показывает все переменные по умолчанию.
6. **echo $** - вызов переменной.
7. **unset** НАЗВАНИЕ ПЕРЕМЕННОЙ – удаление.
8. **grep** – поиск по отдельному слову (env | grep -w НАЗВАНИЕ ПЕРЕМЕННОЙ).

ПРАВА НА ФАЙЛЫ:

1. la -l (файлы начинаться с **“-”**, **“l”**- символическая ссылка, **“d”**- директория(папка).
2. Архивы – красные + двойное расширение.

Команда **chmod :**

chmod ugo+x (readme.txt - полный путь к файлу)

: u (user) - 000 хозяин(создатель) файла, g (grop) - 000 группа, o(other) - 000 другие.

**(-rw)(-r-)(r--)**

1 – исполнение(x)2 – запись(w)3 - запись и исполнение(w,x)4 – чтение(r)5 - чтение и исполнение(r,x)

6 - чтение и запись(r,w)7 - чтение, запись, исполнение(r,w,x)

0 – ничего не может делать с файлом.

ПРИМЕРЫ:

Первая цифра - user, вторая цифра – grop, третья цифра – other.

chmod 644 filename.txt

chmod 700 script.sh

chmod 600 filename.txt

**Tasks Linux / Nov 30, 2022**

**Документация** - первоисточник для любого разработчика

Сегодня большинство серверов работает под управлением ОС на базе ядра Linux.

**Типовые задачи для работы на Linux-сервере**

1. Диагностика неисправностей
2. Развертывание проектов
3. Управление ПО (установка, удаление, настройка)
4. Резервное копирование

**Скрипты**

Способ автоматизации рутинных задач

Языки скриптов (пример):

1. Язык оболочки BASH
2. Python
3. Perl
4. PHP

**Варианты запуска скриптов (ручным способом)**

1. Через явное указание интерпретатора
   1. bash script.sh
2. Через привязку к интерпретатору в самом файле
   1. #!/bin/bash (первая строка)
   2. chmod +x script.sh
   3. ./script.sh

**SSH/RSA-ключи**

* Приватный (private)
* Публичный (public)

Приватный ключ хранится на моем компьютер.

**RSA** алгоритм генерации пары ключей (ассиметричный криптографический алгоритм)

**Диагностика публичного на GitHub (на своем компьютере)**

ssh -T git@github.com

Если на выходе выполучается Permission Denied - это говорит о том, что публичный ключ не выгружен на GitHub.

**Как посмотреть свой публичный ключ?**

cat ~/.ssh/id\_rsa.pub

**Как посмотреть содержимое каталога ~/.ssh**

ls -lah ~/.ssh

**Как сгенерировать пару RSA/RSA-ключей**

(если этого не делали ранее и команда выше не показывает файл id\_rsa.pub ИНАЧЕ НЕ ДЕЛАТЬ!!!)

ДЕЛАЕТСЯ В НАЧАЛЕ КАРЬЕРЫ

ssh-keygen -t rsa

**Как посмотреть конфигурацию git на своем компьютере?**

git config --global -l

**Как скопировать свою пару ключей на свои другие компьютеры?**

~/.ssh - каталог (находится в дом/директории), в котором находится пара RSA-ключей

RSA-ключи - обычные текстовые файлы

**Вопросы по GIT**

**Как посмотреть последний коммит?**

git show

**Как клонировать (выгрузить) репо на компьютер?**

git clone скопированная\_ссылка

**Как удалить каталог?**

rm -rf название\_каталога

rm -rf dir\_\*

**Примеры удаления файлов**

rm filename

rm \*.html

rm \*.css

rm \*.txt

**Как посмотреть статистику по дискам?**

df -h (disk free, human-readable)

**Как посмотреть информацию о занимаемом пространстве?**

du -hs (текущий каталог)

du -hs какой\_то\_каталог

&& - логическое И (AND)

mkdir dir1 && some\_command

mkd dir1; some\_command

~~mkdir dir1 | some\_command~~