

Программирование на Bash: Разбор скриптов и написание своих скриптов. Linter. Shell check



Артём
Палецкий



Артём Палецкий

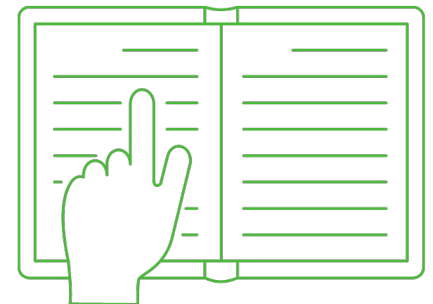
Ведущий инженер, InfoWatch



Предисловие

На этом занятии мы:

- подведем итог созданию скриптов `bash`;
- рассмотрим примеры скриптов;
- узнаем об оформлении скриптов.



План занятия

1. [Предисловие](#)
2. [ARPing](#)
3. [Работа с архивами](#)
4. [Работа с веб-страницами](#)
5. [Xargs](#)
6. [Работа с устройствами](#)
7. [Работа со строками](#)
8. [Стиль оформления](#)
9. [Итоги](#)
10. [Домашнее задание](#)



ARPing



Определение хостов с помощью arping

arping – утилита, позволяющая создавать **icmp**- и **arp**-пакеты.

➔ Рассмотрим скрипт, выполняющий перечисление хостов сегмента сети.

Для того, чтобы перебрать все хосты в заданной подсети, будет использовать **двойной цикл**.



Работа с архивами



Просмотр файла в архиве

➡ Рассмотрим скрипт, выполняющий заданные команды над файлами внутри архива.

Для того, чтобы выбирать нужные режимы работы с архивами, используем оператор `case`.

➡ Рассмотрим применение `eval` для выполнения команд.



Работа с веб-страницами



Поиск ссылок на странице

➡ Рассмотрим применение браузера `lynx`, выполняющий заданные команды над файлами внутри архива.

Для того, чтобы выбирать нужные режимы работы, используем оператор `case`.



Xargs



Заккрытие процесса по имени

- ➔ Рассмотрим скрипт, аналогичный утилите `killall`.
- ➔ Рассмотрим применение:
 - `xargs`;
 - `awk`;
 - подстановку `{ }`.



Работа с устройствами



Создание RAM-диска

➡ Рассмотрим скрипт, создающий диск в [RAM](#).

Для того, чтобы создать и работать с новым устройством, используем команды монтирования диска и создания ФС.



Работа со строками



Генерация случайного пароля

➡ Рассмотрим скрипт, создающий случайный пароль.

Для того, чтобы создать случайную строку, используем генерацию случайных чисел и цикл `while`.



Стиль оформления



Lint. ShellCheck.

➡ Рассмотрим стандарты оформления исходного кода программ (скриптов).

Для проверки исходного кода будем использовать утилиту [Lint. ShellCheck.](#)

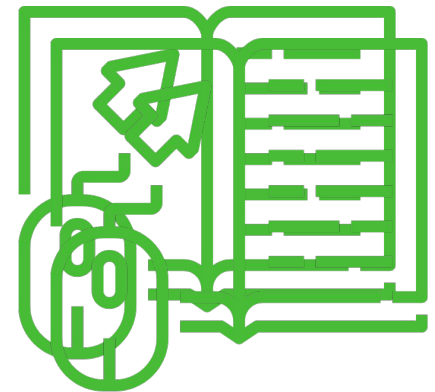


Итоги

Итоги

Сегодня мы рассмотрели различные скрипты, работающие с:

- сетью;
- устройствами;
- строками;
- архивами.





Домашнее задание

Домашнее задание

Давайте посмотрим ваше [домашнее задание](#).

- Вопросы по домашней работе задавайте **в чате** мессенджера Slack.
- Задачи можно сдавать **по частям**.
- Зачёт по домашней работе проставляется после того, как **приняты все задачи**.

**Задавайте вопросы и
пишите отзыв о лекции!**

Артем Палецкий