

СОДЕРЖАНИЕ:

1. Описание
2. Запуск
3. Описание файла с тестами
4. Пример заполнения тестового файла
5. Дополнительные параметры тестового файла
6. Описание лог файлов
7. Описание работы скрипта `./drivedetect/mount_drive`
8. Тестирование работы скрипта (скриншоты)

1) Описание

Скрипт предназначен для следующих проверок

1. Статус служб
2. Присутствия пользователей в системе
3. Наличия пакетов
4. Существование файлов
5. Поиск строки в файлах на сменном носителе

2) Запуск

1. Скопировать файлы в систему
2. Перейти в директорию со скриптом `test`
3. Установить `chmod +x ./test`
4. Создать файл с тестами (либо отредактировать файл «`testfile`» с тестовыми проверками)
5. Запустить файл из каталога со скриптом `./test testfile`
6. Дождаться выполнения скрипта (вывод будет отображаться на экране)
7. Подробная информация о выполнении тестов содержится в директории `./log` (относительно выполняемого скрипта `test`)

Дополнительная информация:

Скрипт запускается только с правами `root` пользователя (под пользователем `root`, либо с использованием `sudo`, если пользователь добавлен в `/etc/sudoers`)

Устанавливать права выполнения на скрипт монтирования сменных носителей (`./drivedetect/mount_drive`) – не обязательно, `chmod +x` для этого скрипта, устанавливается в процессе запуска основного скрипта.

Возможен запуск не из директории скрипта, а с указанием полного пути до скрипта и тестового файла

Запуск скрипта `test` без указания тестового файла отобразит предупреждение : `warning :`
Передайте путь к файлу с тестами: `./test <test_file>`

Если указать неверное имя файла, отобразится предупреждение : `warning :` Файл `filename` не найден

Директория `./log` создается автоматически и имеет права пользователя, запустившего скрипт (в том числе файлы логов)

3) Описание файла с тестами

Файл со списком тестов может содержать любое количество строк (тестов).

Строка должна содержать название теста и атрибут (через пробел).

Запуск тестов без атрибутов – будет выводить сообщение об отсутствии передаваемого параметра для конкретного теста.

Файл может выполнять следующие тесты:

1. `check_status_service` – проверка статуса службы
2. `check_user_exist` – проверка наличия пользователя в системе
3. `check_packet_exist` – проверка установлен ли пакет в системе
4. `check_file_exist` – проверка существования файла по указанному пути
5. `check_string_drive` – проверка наличия файлов и строки поиска в фалах

4) Пример заполнения тестового файла

```
check_status_service cron
check_user_exist username
check_packet_exist tar
check_file_exist /etc/hosts
check_string_drive CFG_TEST somefile
check_string_drive CFG_TEST
```

Примечание:

Тест `check_string_drive` может принимать как один, так и два атрибута.

В первом случае `check_string_drive CFG_TEST somefile` будет производиться поиск строки `CFG_TEST` только в файлах с именем `somefile`

Во втором случае `check_string_drive CFG_TEST` будет производиться поиск строки `CFG_TEST` во всех файлах.

Данный тест производит поиск во всех директориях, всех разделов, подключаемого сменного носителя.

Скрипт определяет, как `usb flash drive` так и `usb hdd`, с одним или более разделами на устройстве, а также отформатированные без таблицы разделов (стандартное форматирование windows для флешек)

5) Дополнительные параметры тестового файла

Для удобства чтения вывода на экран и в лог-файл, предусмотрено разделение групп тестов параметром `line` (строка содержащая текст «`line`» между названий тестов, в файле тестов)

Например, файл может выглядеть так:

```
check_status_service cron
line
check_user_exist username
```

Для установки ожидания перед выполняемым тестом, можно установить паузу строкой «`wait 10`» в файле с тестами. В качестве атрибута передается количество секунд. Допустима строка «`wait`» без указания атрибута, тогда пауза составит 5 секунд.

Во время выполнения, количество оставшихся секунд выводится в консоль.

Такой параметр нужен, например, для того, чтобы вставить `usb flash|hdd` и сразу запустить тест, так как без ожидания, может не успеть произойти инициализация устройства.

6) Описание лог файлов

В директории `./log` содержатся следующие файлы:

1. `all-tests-status.log` – статусы выполненных тестов, разделённые на блоки, для каждого запуска скрипта. Блок начинается с даты и времени запуска, указанием пути и имени тестового файла. Блок заканчивается датой и временем, и сообщением об окончании выполнения тестов.
2. `files-lines.log` – появляется только при запуске теста для поиска на сменных носителях. Разделен на блоки для каждого теста `check_string_drive` из файла. Содержит информацию о времени запуска, искомую строку, имя и путь к файлу, номера строк и их содержимое в файле.
3. `mount.log` – также появляется только при запуске теста для поиска на сменных носителях. Содержит информацию о дате монтирования разделов носителя, список устройств и разделов, выводит сообщения об возникших ошибках (например, о невозможности монтирования неизвестной файловой системы)

7) Описание работы скрипта `./drivedetect/mount_drive`

Скрипт монтирования сменных носителей `./drivedetect/mount_drive` может использоваться самостоятельно (например для других скриптов).

Скрипт принимает следующие атрибуты:

1. `mnt` – создание директорий в `/mnt` и монтирование разделов устройства
2. `umn` – размонтирование разделов устройства
3. `del` – удаление директорий в `/mnt` для обнаруженного устройства

При выполнении с атрибутом `mnt` – скрипт возвращает пути до директорий в `/mnt` для всех успешно смонтированных разделов устройства.

В случае автоматического монтирования носителя (например, при использовании графической оболочки `linux`), скрипт возвращает пути (например, в `Debian` с `Gnome`, пути автоматического монтирования выглядят как `/media/username/id-drive`)

Примечание (для использования отдельно от скрипта теста):

Скрипт видит только последнее подключенное устройство. Рекомендуется подключать дополнительный носитель или менять, после использования атрибутов `umn` и `del` (`mount_drive umn del`) иначе размонтирование и удаление директорий придется производить вручную.

8) Тестирование работы скрипта (скриншоты)

Проверка работы скрипта производилась в следующем окружении:

Operating System: `Debian GNU/Linux 12 (bookworm)`
Kernel: `Linux 6.1.0-20-amd64`

Тестовый файл с тестами – содержит все возможные сценарии.

1. Все тесты с аргументами
2. Все тесты без аргументов (для отображения сообщения)
3. Строку `line`
4. Строку `wait`
5. Пустую строку
6. Неизвестный тест

Содержание тестового тест-файла:

```
check_status_service cron
check_status_service nginx
check_status_service
line
check_user_exist evgenii
check_user_exist username
check_user_exist
line
check_packet_exist tar
check_packet_exist postgres
check_packet_exist
line
check_file_exist /etc/hosts
check_file_exist /etc/nofilename
check_file_exist
line
wait 7
check_string_drive CFG_TEST somefile
check_string_drive CFG_TEST
check_string_drive loremipsum
check_string_drive
line

unknown_test_name
line
```

Скриншот выполнения тестов в консоли:

```
root@i5-debian:/home/evgenii/tests_linux# ./test testfile
[PASS] сервис cron запущен
[FAIL] сервис nginx не запущен
[SKIP] для теста check_status_service название сервиса не указано
-----
[PASS] пользователь evgenii есть в системе
[FAIL] пользователь username отсутствует
[SKIP] для теста check_user_exist имя пользователя не указано
-----
[PASS] пакет tar установлен
[FAIL] пакет postgres не установлен
[SKIP] для теста check_packet_exist название пакета не указано
-----
[PASS] файл /etc/hosts существует
[FAIL] файл /etc/nofilename не найден
[SKIP] для теста check_file_exist путь к файлу не указан
-----
[INFO] Ожидание 7 секунд...
[INFO] Прошло 7 секунд, продолжаем выполнение скрипта.
[PASS] файлы с именем somefile с содержимым CFG_TEST обнаружены
[PASS] файлы с любым именем с содержимым CFG_TEST обнаружены
[FAIL] отсутствуют файлы с любым именем с содержимым loremipsum
[SKIP] для теста check_string_drive не передана строка поиска
-----
[SKIP] неизвестный тест:
[SKIP] неизвестный тест: unknown_test_name
-----
[INFO] Подробная информация в /home/evgenii/tests_linux/log/
root@i5-debian:/home/evgenii/tests_linux#
```

Скриншот ./log/all-tests-status.log:

```
2024-04-23 09:15:31 - тесты запущены из файла /home/evgenii/tests_linux/testfile
[PASS] сервис cron запущен
[FAIL] сервис nginx не запущен
[SKIP] для теста check_status_service название сервиса не указано
-----
[PASS] пользователь evgenii есть в системе
[FAIL] пользователь username отсутствует
[SKIP] для теста check_user_exist имя пользователя не указано
-----
[PASS] пакет tar установлен
[FAIL] пакет postgres не установлен
[SKIP] для теста check_packet_exist название пакета не указано
-----
[PASS] файл /etc/hosts существует
[FAIL] файл /etc/nofilename не найден
[SKIP] для теста check_file_exist путь к файлу не указан
-----
[PASS] файлы с именем somefile с содержимым CFG_TEST обнаружены
[PASS] файлы с любым именем с содержимым CFG_TEST обнаружены
[FAIL] отсутствуют файлы с любым именем с содержимым loremipsum
[SKIP] для теста check_string_drive не передана строка поиска
-----
[SKIP] неизвестный тест:
[SKIP] неизвестный тест: unknown_test_name
-----
2024-04-23 09:15:31 - окончание выполнения тестов
```

1 Помощь

2 Сохранить

3 Блок

4 Замена

5 К

Скриншот ./log/files-lines.log:

```
mc [root@i5-debian]:/home/evgenii/tests_linux/log
/home/evgenii/tests_linux/log/files-lines.log  [----]  0 L:[ 1+13 14/ 14] *(1084/1084b) <EOF>
2024-04-23 09:19:21 строка CFG_TEST (файлы с именем somefile) найдена в файле: /mnt/sdc2/dir1/somefile
- номер строки: 1 Содержимое строки: CFG_TEST=one
- номер строки: 4 Содержимое строки: CFG_TEST=two
*
2024-04-23 09:19:21 строка CFG_TEST (файлы с любым именем) найдена в файле: /mnt/sdc2/dir1/somefile
- номер строки: 1 Содержимое строки: CFG_TEST=one
- номер строки: 4 Содержимое строки: CFG_TEST=two
2024-04-23 09:19:21 строка CFG_TEST (файлы с любым именем) найдена в файле: /mnt/sdc2/dir2/dir3/file
- номер строки: 1 Содержимое строки: CFG_TEST=one
- номер строки: 4 Содержимое строки: CFG_TEST=two
2024-04-23 09:19:21 строка CFG_TEST (файлы с любым именем) найдена в файле: /mnt/sdc3/filename
- номер строки: 3 Содержимое строки: CFG_TEST=two
*
```

Скриншот ./log/mount.log:

```
Вт 23 апр 2024 09:19:29 +05
[INFO] Обнаружено устройство - sdc
[INFO] Все устройства sdc:
- sdc
- sdc1
- sdc2
- sdc3
- sdc4
[INFO] Устройство содержит следующие разделы:
- sdc1
- sdc2
- sdc3
- sdc4
Устройство sdc2 успешно примонтировано
Устройство sdc3 успешно примонтировано
Устройство sdc4 успешно примонтировано
[WARNING] В процессе выполнения монтирования разделов возникли следующие ошибки:
- mount: /mnt/sdc1: wrong fs type, bad option, bad superblock on /dev/sdc1, missing code

Вт 23 апр 2024 09:19:29 +05
[INFO] Обнаружено устройство - sdc
[INFO] Все устройства sdc:
- sdc
- sdc1
- sdc2
- sdc3
- sdc4
[INFO] Устройство содержит следующие разделы:
- sdc1
- sdc2
- sdc3
- sdc4
[WARNING] В процессе размонтирования разделов возникли следующие ошибки:
- umount: /dev/sdc1: not mounted.
```

1Помощь2Сохранить3Блок4Замена

Ошибки связаны с тем, что флешка содержит 4 раздела (для теста). Три из них fat, fat32, ntfs, но один раздел sdc1 не отформатирован.