



JBFW 2 минуты назад

Подключаем WiFi вручную

🕒 6 мин

👁 17

Настройка Linux*

И снова вспомню времена древние, староглиняные: когда-то настройка сети в UNIX-like OS была делом, требующим сложных технических знаний. Нужно было зайти в консоль под правами рутовыми, да указать вручную адрес IP для интерфейса сетевого. Примерно так:

```
ifconfig eth0 add inet 192.168.1.10 netmask 255.255.255.0 up
```

Или, ежели админ сети сподобился, да настроил сервер DHCP - то так:

```
ifconfig eth0 up  
dhclient eth0
```

Да еще не забыть указать адрес сервера DNS:

```
echo "nameserver 8.8.8.8" > /etc/resolv.conf
```

А ежели надобность была чтобы оно при перезагрузке сохранялось - то записать эти заклинания в файл типа /etc/rc.local на память долгую.

Теперь-то другое дело: воткнул шнурок в разъем - сеть и появилась, как по щучьему велению. И даже WiFi - нажал мышкой на иконку, выбрал название сети, ввел пароль - она и подключилась. Красота, лепота какая! Но есть нюанс...

За всю эту красоту отвечает теперь NetworkManager (я про Ubuntu и подобные). Он стартует при запуске компьютера, отслеживает подключения, управляет ими. Да и с /etc/resolv.conf все не так просто, теперь этим управляет другой демон, перехватывая обращения к DNS.

Всё это хорошо пока вы не выходите за рамки предустановленного и настроенного десктопа, в смысле, программной среды.

Но стоит что-то поменять - и оказывается, что теперь некуда тыкать мышкой, потому что иконки нет. А нет ее потому что апплет, который за нее отвечает, не рассчитан на другую среду, или "не удается подключиться к Dbus", или оно как бы работает - но почему-то не так как ожидается.

И как бы есть даже интерфейс командной строки nmcli, который взаимодействует с NetworkManager, но с ним тоже всё не так просто, потому что вся эта довольно сложная система теперь работает в нештатном режиме.

Однако, всё это по-прежнему можно легко настроить вручную, и даже автоматизировать, без использования сложных программных пакетов.

(отдельно отмечу, что не стоит пытаться объять необъятное и писать свой собственный

РЕКЛАМА

**Ночь в дата-центре**

Ваша самая необычная смена

Итак, подключаемся вручную.

Первым делом отключим NetworkManager чтобы он не мешал:

```
systemctl stop NetworkManager  
systemctl disable NetworkManager
```

Что касается обычных сетевых интерфейсов - тут мало что изменилось, разве что вместо ifconfig теперь модно использовать ip:

```
ip link set eth0 up  
ip addr add 192.168.1.10 dev eth0
```

А для работы с WiFi используются другие инструменты (iwconfig или iw):

```
iwconfig  
lo no wireless extensions.  
wlp2s0 IEEE 802.11 ESSID:off/any  
Mode:Managed Access Point: Not-Associated Tx-Power=22 dBm  
Retry short limit:7 RTS thr:off Fragment thr:off  
Power Management:on
```

В данном случае найдены два интерфейса, локальный без поддержки WiFi и wlp2s0
Убедимся, что он поднят и попробуем поискать сети:

```
ip link set wlp2s0 up  
iwlist wlp2s0 scan | grep ESSID
```

Получаем список доступных сетей. Скорее всего ваша сеть использует для защиты WPA/WPA2, другие в наше время экзотика, поэтому понадобится wpa_supplicant.

Для того чтобы работать с сетью нужно прописать для него конфигурационный файл, примерно с таким содержанием:

```
network={  
    ssid="[network ssid]"  
    psk="[the passphrase]"  
    priority=1  
}
```

Вообще говоря, в одном файле можно держать настройки для разных сетей, но если не хочется потом сюрпризов с неожиданными переключениями - лучше ограничиться одной, и просто менять ее или используемые конфиги при необходимости.

Запускаем сам wpa_supplicant и dhclient:

```
wpa_supplicant -B -i wlp2s0 -c /etc/wpa_supplicant/wpa_supplicant_XXXX.conf  
dhclient wlp2s0
```

В типовом случае этого будет достаточно. Если всё было правильно - компьютер подключится к сети и получит адрес.



Ночь в дата-центре

Ваша самая необычная смена

Для создания графического интерфейса за основу взята довольно старинная парочка tcl/tk, в отличии от более современных вариантов не требовательная к ресурсам, к тому же прекрасно интегрирующаяся в обычные shell-скрипты. Если вдруг этих пакетов не установлено - всегда можно их добавить, там всего-то пара мегабайт:

```
apt install tcl tk
```

Чтобы не писать свой NetworkManager - ограничимся простым скриптом, который делает вон то же самое что написано выше, только информацию от пользователя получает через окошки.

```
#!/bin/bash

# окно ввода пароля
ask_pass() {

    getpwd=""
    package require Tk

    # настройки окна
    wm title . \"$1\"
    wm geometry . +300+50
    wm protocol . WM_DELETE_WINDOW { exit }

    # сообщение и строка ввода со звездочками
    label .label -text \"$1:\" -padx 10 -pady 10
    pack .label -side top
    entry .entry -width 30 -show * -textvariable userInput
    pack .entry -side top -padx 10 -pady 5
    set userInput \"\"

    # функция при нажатии на кнопку
    proc onOk {} {
        global userInput
        puts stdout \"$userInput\"
        exit
    }

    # создание кнопки и размещение ее в окне
    button .okButton -text \"OK\" -command onOk -padx 10 -pady 5
    pack .okButton -side top -padx 10 -pady 10

    # ожидание ввода
    vwait userInput
    "
    echo \"$getpwd\" | wish
}

# окно сообщения
message() {
```



```

# настройки окна
wm title . \"$1\"
wm geometry . +300+50
wm minsize . 300x150
wm protocol . WM_DELETE_WINDOW { exit }

# сообщение
label .label -text \"$1\" -padx 10 -pady 30
pack .label -side top

# функция при нажатии на кнопку
proc onOk {} {
    exit
}

# создание кнопки и размещение ее в окне
button .okButton -text \"OK\" -command onOk -padx 10 -pady 5
pack .okButton -side top -padx 10 -pady 5

"
    echo "$str" | wish
}

# проверяем, скрипт запускается от имени суперпользователя или нет
if [[ $EUID -ne 0 ]]; then

    # проверяем, делалось ли sudo
    sudo -n true > /dev/null 2>&1
    if [ $? -ne 0 ] ; then
        passwd=`ask_pass "User password"`

        if [ ! -n "$passwd" ] ; then
            message "No password!"
            exit 1
        fi

        echo $passwd | sudo -S true > /dev/null 2>&1
        if [ $? -ne 0 ] ; then
            message "Incorrect password!"
            exit 2
        fi
    fi
fi

# получаем список интерфейсов с поддержкой WiFi
interfaces=$(iwconfig 2>/dev/null | awk '/IEEE/ {print $1 "=" $4}')
if [[ -z "$interfaces" ]]; then
    echo "No WiFi found" >&2
    exit 3
fi

# создаем временный файл для списка интерфейсов и AP

```



Ночь в дата-центре

Ваша самая необычная смена

```

for iface in $interfaces; do
    echo "$iface" >> "$temp_file"
done

iface=''

# вывод окна списка интерфейсов
select_iface() {

    getiface="
package require Tk
wm title . \"Select interface\"
wm geometry . +300+50
wm protocol . WM_DELETE_WINDOW { exit }

label .l1 -text \"Select interface\"
pack .l1 -padx 10 -pady 5

listbox .list1 -width 50 -height 3
pack .list1 -padx 10 -pady 5

# загружаем данные из временного файла
set fd [open \"$temp_file\" r]
set lines [split [read $fd] \"\\n\"]
close $fd

foreach line $lines {
    if {[regexp {(\\w+)=ESSID:([\"\\w]+)} $line match iface essid]} {
        .list1 insert end \"$iface ($essid)\"
    }
}

# кнопка выбора интерфейса
button .ok -text \"Select interface\" -command {
    set selection [.list1 get [.list1 curselection]]
    if {[regexp {(\\w+) } $selection match iface]} {
        set selected_iface $iface
        puts stdout \"$selected_iface\"
        exit
    }
}
pack .ok -padx 10 -pady 5

set selected_iface \"\"

vwait selected_iface

"

    echo "$getiface" | wish

```



Ночь в дата-центре

Ваша самая необычная смена

```

# выбираем интерфейс
iface=`select_iface`

# вывод окна списка сетей
select_network() {

getnet=""
package require Tk
wm title . \"Select network\"
wm geometry . +300+50
wm protocol . WM_DELETE_WINDOW { exit }

label .l1 -text \"Select network\"
pack .l1 -padx 10 -pady 5

listbox .list -width 50 -height 20
pack .list -padx 10 -pady 5

# Загружаем данные из временного файла
set fd [open \"$temp_file\" r]
set lines [split [read $fd] \"\\n\"]
close $fd

foreach line $lines {
    if {[regexp {(\\w+)} $line match essid]} {
        .list insert end \"$essid\"
    }
}

button .ok -text \"Select\" -command {
    set selection [.list get [.list curselection]]
    set selected_net $selection
    puts stdout \"$selected_net\"
    exit
}

pack .ok -padx 10 -pady 5

set selected_net \"\"

vwait selected_net

"

echo "$getnet" | wish

}

# функция для сканирования сетей
scan_networks() {
    sudo ip link set $1 up
    echo -n "" > "$temp_file"
}

```



Ночь в дата-центре

Ваша самая необычная смена

```

}

# если интерфейс выбран - сканируем сети и выбираем из найденных
if [ -n "$iface" ] ; then

    scan_networks $iface

    net=`select_network`

    if [ -n "$net" ] ; then
        key=`ask_pass "WiFi password"`
        if [ -n "$key" ] ; then

            # Создаем конфигурацию для wpa_supplicant
            conf=""
            update_config=1

            network={
                ssid="$net\"
                psk="$key\"
            }
            "

            sudo mkdir -p /etc/wpa_supplicant
            file=/etc/wpa_supplicant/wpa_supplicant_$iface.conf

            # шаманство с созданием файла
            sudo touch $file
            sudo chmod 666 $file
            sudo echo "$conf" > $file
            sudo chmod 644 $file

            # перезапускаем wpa_supplicant
            pid=`ps ax| grep "wpa_supplicant" | grep -v grep | grep "$iface" | awk '{print
            if [ "x$pid" != "x" ] ; then
                sudo kill $pid
            fi

            pid=`ps ax| grep "dhclient $iface" | grep -v grep | awk '{print $1}'`
            if [ "x$pid" != "x" ] ; then
                sudo kill $pid
            fi

            sudo wpa_supplicant -B -i "$iface" -c /etc/wpa_supplicant/wpa_supplicant_$iface

            # запрашиваем IP-адрес через DHCP
            sudo dhclient "$iface"

            msg=`ip addr show dev $iface | grep inet | awk '{print $1 " " $2 }'`
            message "$msg"

        else

```



Ночь в дата-центре

Ваша самая необычная смена

```

else
    message "No network selected!"
fi
else
    message "No interface selected!"
fi

exit

#=====

```

Скрипт проверяет, под кем он запущен, если под обычным пользователем - запрашивает пароль для sudo, затем получает список интерфейсов с WiFi (их может быть несколько), после выбора сканирует доступные сети, предлагает выбрать, потом формирует конфиг для данного интерфейса и запускает wpa_supplicant. Если все прошло хорошо - показывает IP-адреса на этом интерфейсе.

Для того чтобы не запускать всё это каждый раз после перезагрузки - есть другой скрипт, который должен запускаться от рута при старте системы:

```

#!/bin/sh

for i in /etc/wpa_supplicant/wpa_supplicant_*.conf; do
    iface=`echo $i | sed -nE "s/^.*/wpa_supplicant_(.+)\.conf/\1/p"`
    if [ -n "$iface" ] ; then
        ip link set $iface up
        wpa_supplicant -B -i "$iface" -c /etc/wpa_supplicant/wpa_supplicant_${iface}.conf
        dhclient "$iface"
    fi
done

```

Проверяет наличие сохраненных конфигов и запускает wpa_supplicant с ними.

Теги: [wifi](#)

Хабы: [Настройка Linux](#)

Редакторский дайджест

Присылаем лучшие статьи раз в месяц



35

149.7



Ночь в дата-центре

Ваша самая необычная смена

Пользователь

Подписаться



Комментировать

Публикации

ЛУЧШИЕ ЗА СУТКИ

ПОХОЖИЕ



IZaron 21 час назад

Создаем эмулятор Sega Mega Drive на C++



Средний



26 мин



9.8K



+84



132



19 +19



sayas 3 часа назад

На батуте в космос. Разработка средств вывода без участия Роскосмоса



Простой



8 мин



3.6K

Из песочницы



+31



8



19 +19



LeonidFirstov 22 часа назад

Системные ошибки в преподавании иностранных языков



7 мин



12K



+30



46



37 +37



OlegSivchenko 22 часа назад

Негостеприимные красные карлики. Об ультрафиолетовой зоне обитаемости



9 мин



3.1K



+24



21



4 +4



Erwinmal 7 часов назад

Суши-роллы: как традиционное японское блюдо мутировало за тысячи лет, попало в Калифорнию и покорило весь мир?



Ночь в дата-центре

Ваша самая необычная смена

◆ +21 📖 8 💬 2 +2



tsurkan_n 8 часов назад

Надоело постоянно пользоваться телефоном, поэтому написал замену Google Authenticator в виде TUI

👁️ Простой ⌚ 7 мин 👁️ 6.8K

Из песочницы

◆ +18 📖 34 💬 35 +35



oneastok 8 часов назад

Новогодняя подборка образовательных материалов для разработчиков

👁️ Простой ⌚ 4 мин 👁️ 1.5K

Обзор

◆ +17 📖 26 💬 4 +4



Darya_Frolova 7 часов назад

Когда промышленный дизайн становится искусством: идеи Генри Дрейфуса

⌚ 6 мин 👁️ 881

◆ +12 📖 14 💬 2 +2



heavyC1oud 18 часов назад

Исповедь устаревшего

👁️ Простой ⌚ 3 мин 👁️ 4.6K

Ретроспектива

◆ +12 📖 10 💬 3 +3



VetaOne 21 час назад

Как я гифку с помощью ИИ сжимал

👁️ Простой ⌚ 16 мин 👁️ 1.7K

Из песочницы

◆ +10 📖 15 💬 17 +17

Интерком в 2024 году: как нейросети и чат-боты меняют общение с



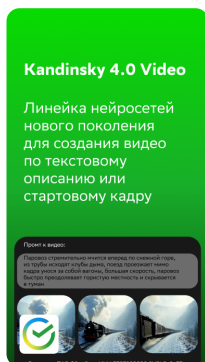
Ночь в дата-центре

Ваша самая необычная смена

Турбо

Показать еще

ИСТОРИИ



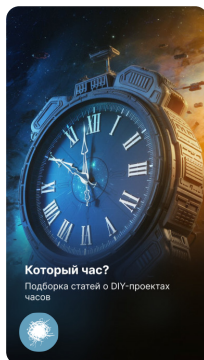
Нейросеть для создания видео



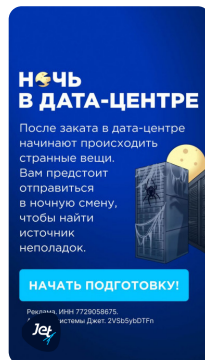
Кто выступит на конференции мечты



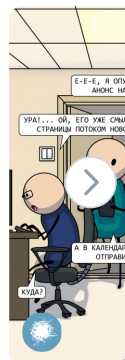
Годнота из блогов компаний



Который час?





Неполадки в ночном дата-центре





Перевернут календарь и добавить со


КУРСЫ

 Systems Analyst Bootcamp: Проектировщик корпоративных информационных систем
16 января 2025 · Systems Education

 Проектирование и реализация очередей в брокерах RabbitMQ и Apache Kafka
18 января 2025 · Systems Education

 Frontend-разработчик
По факту набора · Университет «Синергия»

 Linux уровень 1. Основы администрирования систем Linux (Линукс)
По факту набора · Учебный центр «Микротест»

 Системный администратор
По факту набора · Нетология

[Больше курсов на Хабр Карьере](#)

МИНУТОЧКУ ВНИМАНИЯ



Бези стартеров и акселераций



Сервера на святающей крест во



Фиброконвергентная среда



Ночь в дата-центре

Ваша самая необычная смена

БЛИЖАЙШИЕ СОБЫТИЯ



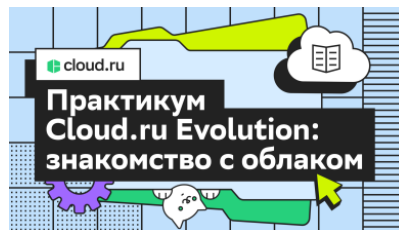
16 января

Вебинар «Практикум Cloud.ru Evolution: знакомство с облаком»

Онлайн

Разработка

[Больше событий в календаре](#)



Хабр



[Настройка языка](#)

[Техническая поддержка](#)

© 2006–2025, Habr



Ночь в дата-центре

Ваша самая необычная смена