

# 利用GitHub ACTIONS 在线云编译大雕的OpenWRT固件

📅 2019年12月3日 💬 0条评论 👁 4.04k次阅读 🍷 1人点赞 👤 博主

有很多朋友都想编译一个合适自己使用的OpenWRT固件，这样的好处就是只保留了自己需要使用的插件，非常精简，那么就需要自己搭建一个编译环境。但是并不是每个人都会搭建，现在有一个很便捷的方案，就是利用GitHub ACTIONS 在线云编译。一起来看看吧：

## 一、注册GitHub账号

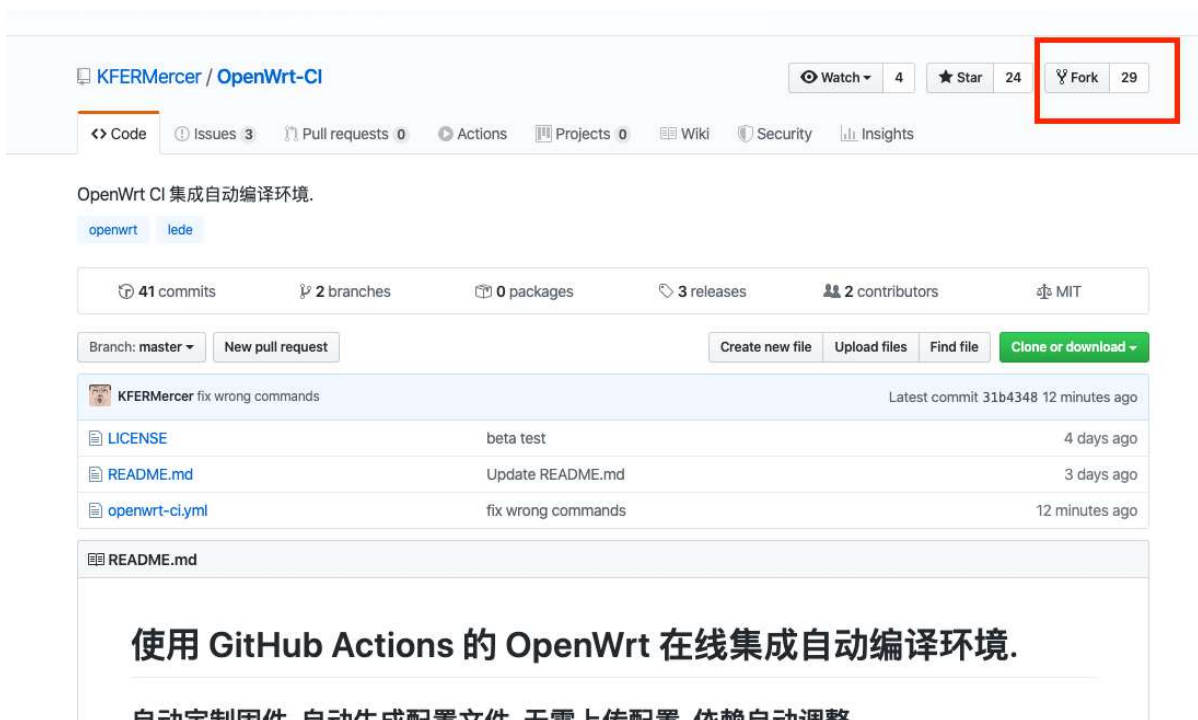
官网地址：<https://github.com/>

## 二、激活GitHub ACTIONS功能（此步骤已经不需要做，直接到项目里面点击Action选项卡，就可以跳转到激活页面了）

激活地址：<https://github.com/features/actions/signup>

## 三、fork 一建脚本项目

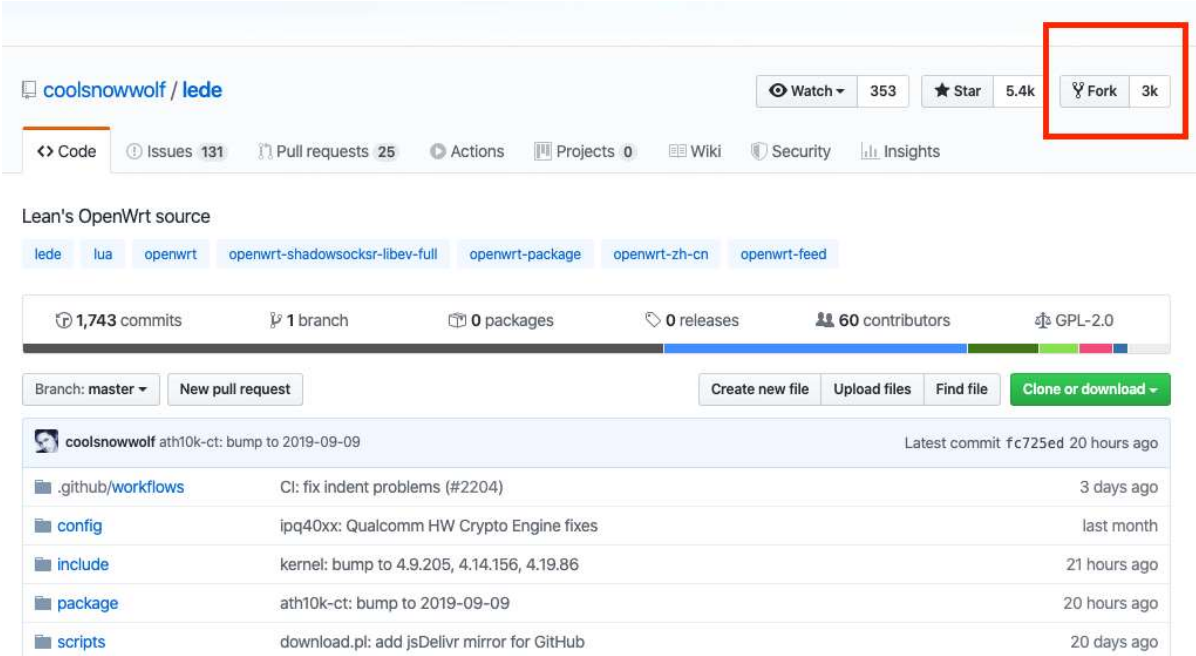
<https://github.com/KFERMercer/OpenWrt-CI>



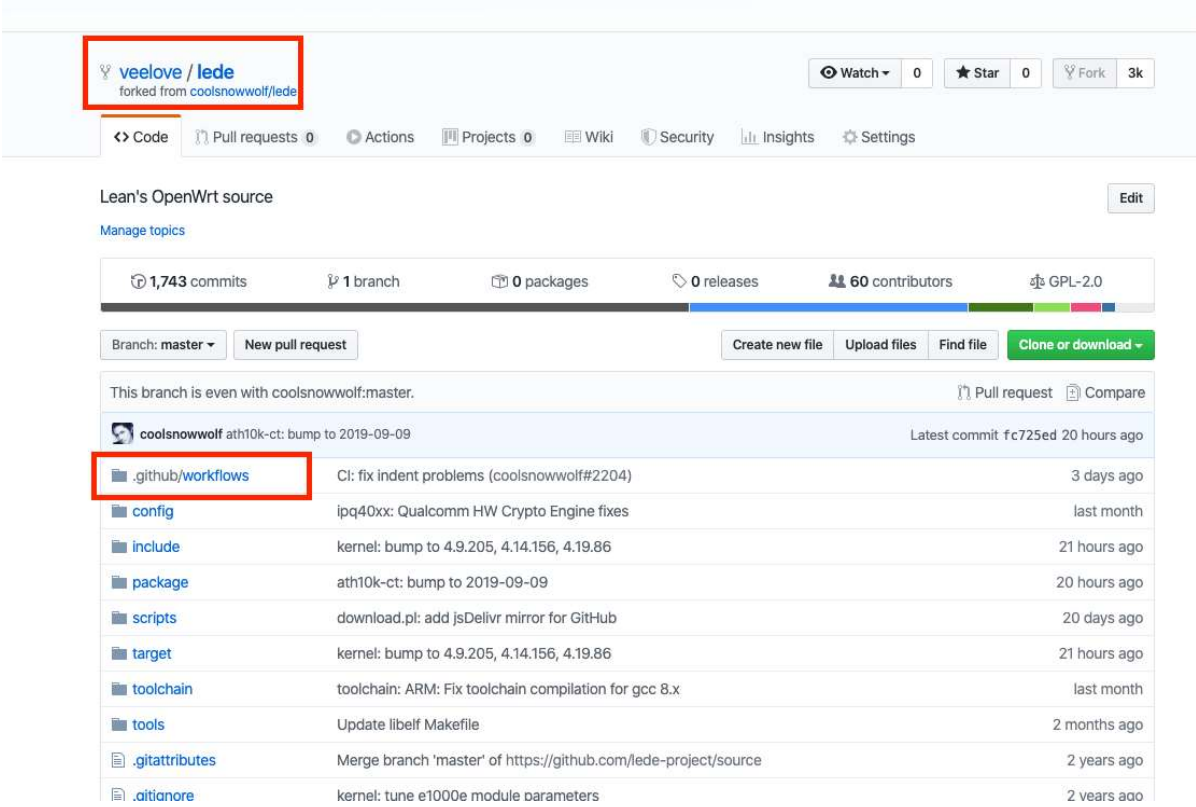
## 四、fork L大的LEDE源码

<https://github.com/coolsnowwolf/lede>

能不能继续对我哭

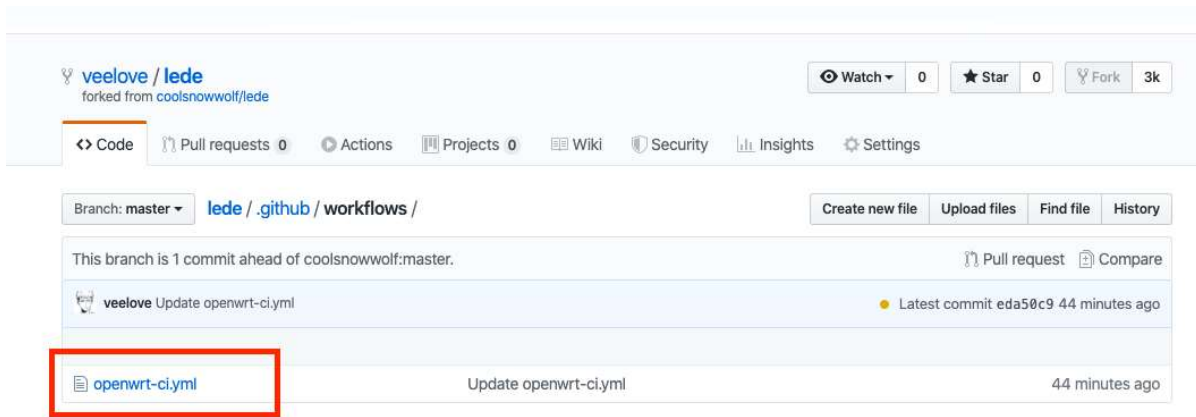


五、进入L大源码目录.github/workflows/



六、打开openwrt-ci.yml，编辑代码（如需要添加其他的主题或者插件，可参考第九和第十条）

能不能继续对我哭



```
#  
# This is free software, lisencc use MIT.  
#  
# Copyright (C) 2019 P3TERX <https://p3terx.com>  
# Copyright (C) 2019 KFERMercer <KFER.Mercer@gmail.com>  
#  
# <https://github.com/KFERMercer/OpenWrt-CI>  
#
```

```
name: OpenWrt-CI
```

```
on:
```

```
  push:
```

```
    branches:
```

```
      - master
```

```
  # schedule:
```

```
  #   - cron: 0 20 * * *
```

```
  # release:
```

```
  #   types: [published]
```

```
jobs:
```

```
  build:
```

```
    runs-on: ubuntu-latest
```

```
    steps:
```

```
      - name: Checkout
```

```
        uses: actions/checkout@master
```

```
        with:
```

```
          ref: master
```

```
      - name: Initialization environment
```

```
        env:
```

```
          DEBIAN_FRONTEND: noninteractive
```

```
        run: |
```

```
          docker rmi `docker images -q`
```

```
          echo "Deleting files, please wait ..."
```

```
          sudo rm -rf \
```

```
            /usr/share/dotnet \
```

```
            /etc/mysql \
```

```
            /etc/php
```

能不能继续对我哭

```

azul-eclat \
ghc* \
zulu* \
hhvm \
llvm* \
firefox \
google* \
dotnet* \
powershell \
openjdk* \
mysql* \
php*
sudo -E apt-get update
sudo -E apt-get -y install build-essential asciidoc binutils bzip2 gawk gettext git libr
sudo -E apt-get -y autoremove --purge
sudo -E apt-get clean

- name: Update feeds
run: |
    ./scripts/feeds update -a
    ./scripts/feeds install -a

- name: Custom configure file
run: |
    rm -f ./config*
    touch ./config

#
# ===== 固件定制部分 =====
#

#
# 如果不对本区块做出任何编辑，则生成默认配置固件。
#

# 以下为定制化固件选项和说明：
#

#
# 有些插件/选项是默认开启的，如果想要关闭，请参照以下示例进行编写：
#
#
# =====
# | # 取消编译VMware 镜像： |
# | cat >> ./config <<EOF |
# | # CONFIG_VMDK_IMAGES is not set |
# | EOF |
# =====
#

#
# 以下是一些提前准备好的一些插件选项。
# 直接取消注释相应代码块即可应用。 不要取消注释代码块上的汉字说明。
# 如果不需要代码块里的某一项配置，只需要删除相应行。
#
# 如果需要其他插件，请按照示例自行添加。
# 注意，只需添加依赖链顶端的包。如果你需要插件 A，同时 A 依赖 B，即只需要添加 A。
#

```

能不能继续对我哭

```

# 编译x64固件:
# cat >> .config <<EOF
# CONFIG_TARGET_x86=y
# CONFIG_TARGET_x86_64=y
# CONFIG_TARGET_x86_64_Generic=y
# EOF

# 固件压缩:
# cat >> .config <<EOF
# CONFIG_TARGET_IMAGES_GZIP=y
# EOF

# 编译UEFI固件:
# cat >> .config <<EOF
# CONFIG_EFI_IMAGES=y
# EOF

# IPv6支持:
# cat >> .config <<EOF
# CONFIG_PACKAGE_dnsmasq_full_dhcpv6=y
# CONFIG_PACKAGE_ipv6helper=y
# EOF

# 多文件系统支持:
# cat >> .config <<EOF
# CONFIG_PACKAGE_kmod-fs-nfs=y
# CONFIG_PACKAGE_kmod-fs-nfs-common=y
# CONFIG_PACKAGE_kmod-fs-nfs-v3=y
# CONFIG_PACKAGE_kmod-fs-nfs-v4=y
# CONFIG_PACKAGE_kmod-fs-ntfs=y
# CONFIG_PACKAGE_kmod-fs-squashfs=y
# EOF

# USB3.0支持:
# cat >> .config <<EOF
# CONFIG_PACKAGE_kmod-usb-ohci=y
# CONFIG_PACKAGE_kmod-usb-ohci-pci=y
# CONFIG_PACKAGE_kmod-usb2=y
# CONFIG_PACKAGE_kmod-usb2-pci=y
# CONFIG_PACKAGE_kmod-usb3=y
# EOF

# 常用LuCI插件选择:
# cat >> .config <<EOF
# CONFIG_PACKAGE_luci-app-adbyby-plus=y
# CONFIG_PACKAGE_luci-app-aria2=y
# CONFIG_PACKAGE_luci-app-baidupcs-web=y
# CONFIG_PACKAGE_luci-app-docker=y
# CONFIG_PACKAGE_luci-app-frpc=y
# CONFIG_PACKAGE_luci-app-hd-idle=y
# CONFIG_PACKAGE_luci-app-kodexplorer=y
# CONFIG_PACKAGE_luci-app-minidlna=y
# CONFIG_PACKAGE_luci-app-openvpn=y
# CONFIG_PACKAGE_luci-app-openvpn-server=y
# CONFIG_PACKAGE_luci-app-qbittorrent=y
```

能不能继续对我哭

```

# CONFIG_PACKAGE_luci-app-ssr-plus_INCLUDE_Shadowsocks=y
# CONFIG_PACKAGE_luci-app-ssr-plus_INCLUDE_ShadowsocksR_Server=y
# CONFIG_PACKAGE_luci-app-ssr-plus_INCLUDE_ShadowsocksR_Socks=y
# CONFIG_PACKAGE_luci-app-ssr-plus_INCLUDE_V2ray=y
# CONFIG_PACKAGE_luci-app-ttyd=y
# CONFIG_PACKAGE_luci-app-v2ray-server=y
# CONFIG_PACKAGE_luci-app-verysync=y
# CONFIG_PACKAGE_luci-app-webadmin=y
# CONFIG_PACKAGE_luci-app-wireguard=y
# CONFIG_PACKAGE_luci-app-wrtbwmon=y
# EOF

# LuCI主题:
# cat >> .config <<EOF
# CONFIG_PACKAGE_luci-theme-argon=y
# CONFIG_PACKAGE_luci-theme-netgear=y
# EOF

# 常用软件包:
# cat >> .config <<EOF
# CONFIG_PACKAGE_curl=y
# CONFIG_PACKAGE_htop=y
# CONFIG_PACKAGE_nano=y
# CONFIG_PACKAGE_screen=y
# CONFIG_PACKAGE_tree=y
# CONFIG_PACKAGE_vim-fuller=y
# CONFIG_PACKAGE_wget=y
# EOF

# 取消编译VMware 镜像以及镜像填充 ( 不要删除被缩进的注释符号):
# cat >> .config <<EOF
# # CONFIG_TARGET_IMAGES_PAD is not set
# # CONFIG_VMDK_IMAGES is not set
# EOF

#
# =====固件定制部分结束=====
#

sed -i 's/^[ \t]*//g' ./config
make defconfig

- name: Make download
  run: |
    make download -j8
    find dl -size -1024c -exec ls -l {} \;
    find dl -size -1024c -exec rm -f {} \;

- name: Compile firmware
  run: |
    echo -e "$(nproc) thread build."
    make -j$(nproc) V=s

- name: Assemble artifact
  run: |
    rm -rf ./artifact/
    mkdir -p ./artifact/
    mv bin/* ./artifact/

```

能不能继续对我哭

```
- name: Upload artifact
  uses: actions/upload-artifact@master
  with:
    name: OpenWrt firmware
    path: ./artifact/
```

七、找到LEDE项目的ACTIONS，查看编译过程

veelove / lede  
forked from coolsnowwolf/lede

Watch 0 Star 0 Fork 3k

CodePull requests 0ActionsProjects 0WikiSecurityInsightsSettings

Lean's OpenWrt source

1,744 commits1 branch0 packages0 releases60 contributorsGPL-2.0

Branch: masterNew pull requestCreate new fileUpload filesFind fileClone or download

This branch is 1 commit ahead of coolsnowwolf:master.

veelove Update openwrt-ci.ymlLatest commit eda50c9 1 hour ago

.github/workflows	Update openwrt-ci.yml	1 hour ago
config	ipq40xx: Qualcomm HW Crypto Engine fixes	last month
include	kernel: bump to 4.9.205, 4.14.156, 4.19.86	21 hours ago
package	ath10k-ct: bump to 2019-09-09	20 hours ago
scripts	download.pl: add jsDelivr mirror for GitHub	20 days ago

veelove / lede  
forked from coolsnowwolf/lede

CodePull requests 0ActionsProjects 0WikiSecurityInsightsSettings

WorkflowsNew workflow

OpenWrt-CI

OpenWrt-CI  
on: push (eda50c9)

1 hour ago13s

veelove / lede  
forked from coolsnowwolf/lede

CodePull requests 0ActionsProjects 0WikiSecurityInsightsSettings

Update openwrt-ci.yml  
master eda50c9

OpenWrt-CI  
on: push

build

OpenWrt-CI / build  
started 51m 26s ago

Set up job

Checkout

Initialization environment

Update feeds

Custom configure file

Make download

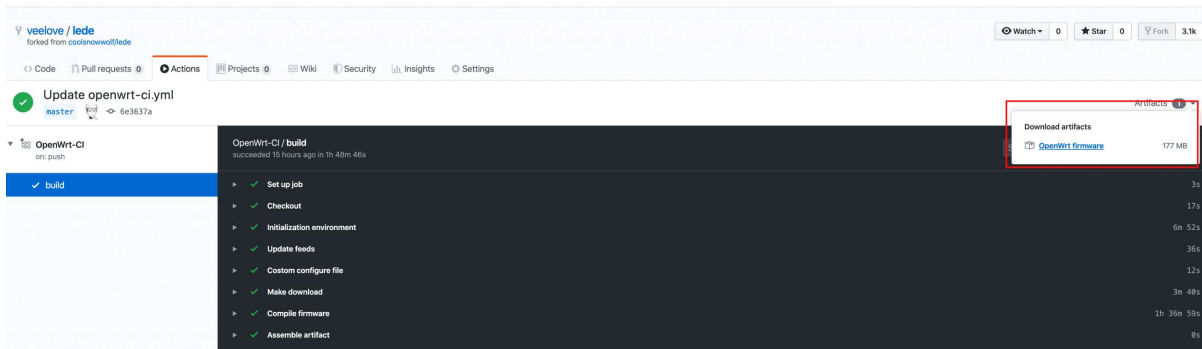
Compile firmware

DHAVE\_CONFIG\_H -I. -I. -I/home/runner/work/lede/lede/build\_dir/toolchain-x86\_64\_gcc-7.4...  
-I/home/runner/work/lede/lede/build\_dir/toolchain-x86\_64\_gcc-7.4.0\_musl/gcc-7.4.0/gcc/..  
I/home/runner/work/lede/lede/build\_dir/toolchain-x86\_64\_gcc-7.4.0\_musl/gcc-7.4.0/gcc/..  
3 g++ -fno-PIE -c -g -O2 -DIN\_GCC -DCROSS\_DIRECTORY\_STRUCTURE -fno-exceptions -fno-r..  
DHAVE\_CONFIG\_H -I. -I. -I/home/runner/work/lede/lede/build\_dir/toolchain-x86\_64\_gcc-7.4...

能不能继续对我哭



## 八、编译完成后下载固件



## 九、自定义插件主题代码

```

- name: 自定义插件主题。
  run: |
    cd package
    mkdir openwrt-packages
    cd openwrt-packages
    git clone https://github.com/openwrt-develop/luci-theme-atmaterial.git
    git clone https://github.com/Ameykyl/luci-app-koolproxyR
    git clone https://github.com/rosywr/luci-theme-rosy
  
```

## 十、LuCi Applications软件包说明

```

LuCI -> Applications -> luci-app-accesscontrol #访问时间控制
LuCI -> Applications -> luci-app-adblock #ADB广告过滤
LuCI -> Applications -> luci-app-adbyby-plus #广告屏蔽大师Plus +
LuCI -> Applications -> luci-app-adbyby #广告过滤大师(已弃)
LuCI -> Applications -> luci-app-adkill #广告过滤(已弃)
LuCI -> Applications -> luci-app-advanced-reboot #Linksys高级重启
LuCI -> Applications -> luci-app-ahcp #支持AHCPd
LuCI -> Applications -> luci-app-aliddns #阿里DDNS客户端
LuCI -> Applications -> luci-app-amule #aMule下载工具
LuCI -> Applications -> luci-app-aria2 #Aria2下载工具
LuCI -> Applications -> luci-app-arpbind #IP/MAC绑定
LuCI -> Applications -> luci-app-asterisk #支持Asterisk电话服务器
LuCI -> Applications -> luci-app-attendedsysupgrade #固件更新升级相关
LuCI -> Applications -> luci-app-autoreboot #支持计划重启
LuCI -> Applications -> luci-app-bcp38 #BCP38网络入口过滤(不确定)
LuCI -> Applications -> luci-app-bird4 #Bird 4(未知)
LuCI -> Applications -> luci-app-bird6 #Bird 6(未知)
LuCI -> Applications -> luci-app-bmx6 #BMX6路由协议
LuCI -> Applications -> luci-app-bmx7 #BMX7路由协议
LuCI -> Applications -> luci-app-caldav #联系人
LuCI -> Applications -> luci-app-cjdnss #加密IPV6网络相关
LuCI -> Applications -> luci-app-clamav #CLamAV杀毒软件
LuCI -> Applications -> luci-app-commands #Shell命令模块
LuCI -> Applications -> luci-app-cloudshark #CloudShark抓包工具
  
```

能不能继续对我哭



LuCI → Applications → luci-app-diag-core #Core 诊断工具

LuCI → Applications → luci-app-dnscrypt-proxy #DNSEncrypt 解决DNS污染

LuCI → Applications → luci-app-dnscrypt-dnsforwarder #DNSForwarder 防DNS污染

LuCI → Applications → luci-app-dnspod #DNSPod

LuCI → Applications → luci-app-dump1090 #民航无线频率(不确定)

LuCI → Applications → luci-app-dynapoint #DynaPoint(未知)

LuCI → Applications → luci-app-e2guardian #Web内容过滤器

LuCI → Applications → luci-app-filetransfer #文件传输

LuCI → Applications → luci-app-firewall #添加防火墙

LuCI → Applications → luci-app-flowoffload #Turbo ACC FLOW转发加速

LuCI → Applications → luci-app-freifunk-diagnostics #freifunk组件 诊断(未知)

LuCI → Applications → luci-app-freifunk-policyrouting #freifunk组件 策略路由(未知)

LuCI → Applications → luci-app-freifunk-widgets #freifunk组件 索引(未知)

LuCI → Applications → luci-app-frpc #内网穿透 Frp

LuCI → Applications → luci-app-fwknopd #Firewall Knock Operator服务器

LuCI → Applications → luci-app-guest-wifi #WiFi访客网络

LuCI → Applications → luci-app-hd-idle #硬盘休眠

LuCI → Applications → luci-app-hnet #Homenet Status家庭网络控制协议

LuCI → Applications → luci-app-kooldns #virtual服务器 ddns替代方案(已弃)

LuCI → Applications → luci-app-koolproxy #KP去广告(已弃)

LuCI → Applications → luci-app-lxc #LXC容器管理

LuCI → Applications → luci-app-meshwizard #网络设置向导

LuCI → Applications → luci-app-minidlna #完全兼容DLNA / UPnP-AV客户端的服务器软件

LuCI → Applications → luci-app-mjpg-streamer #兼容Linux-UVC的摄像头程序

LuCI → Applications → luci-app-mmc-over-gpio #添加SD卡操作界面(已弃)

LuCI → Applications → luci-app-multiwan #多拨虚拟网卡(已弃)

LuCI → Applications → luci-app-mwan #MWAN负载均衡

LuCI → Applications → luci-app-mwan3 #MWAN3分流助手

LuCI → Applications → luci-app-n2n\_v2 #N2N内网穿透 N2N v2 virtual服务

LuCI → Applications → luci-app-nft-qos #QOS流控 Nftables版(已弃)

LuCI → Applications → luci-app-ngrok #Ngrok 内网穿透(已弃)

LuCI → Applications → luci-app-nlbwmon #网络带宽监视器

LuCI → Applications → luci-app-noddos #NodDOS Clients 阻止DDoS攻击

LuCI → Applications → luci-app-ntpc #NTP时间同步服务器

LuCI → Applications → luci-app-ocserv #OpenConnect virtual服务

LuCI → Applications → luci-app-olsr #OLSR配置和状态模块

LuCI → Applications → luci-app-olsr-services #OLSR服务器

LuCI → Applications → luci-app-olsr-viz #OLSR可视化

LuCI → Applications → luci-app-ocserv #OpenConnect virtual服务(已弃)

LuCI → Applications → luci-app-openvirtual #Openvirtual客户端

LuCI → Applications → luci-app-openvirtual-server #易于使用的Openvirtual服务器 Web-UI

LuCI → Applications → luci-app-oscarm #OSCAM服务器(已弃)

LuCI → Applications → luci-app-p910nd #打印服务器模块

LuCI → Applications → luci-app-pagekitee #Pagekite内网穿透客户端

LuCI → Applications → luci-app-polipo #Polipo代理(是一个小型且快速的网页缓存代理)

LuCI → Applications → luci-app-pppoe-relay #PPPoE NAT穿透 点对点协议(PPP)

LuCI → Applications → luci-app-pptp-server #virtual服务器 pptp

LuCI → Applications → luci-app-privoxy #Privoxy网络代理(带过滤无缓存)

LuCI → Applications → luci-app-qos #流量服务质量(QoS)流控

LuCI → Applications → luci-app-radical #CalDAV/CardDAV同步工具

LuCI → Applications → luci-app-ramfree #释放内存

LuCI → Applications → luci-app-rp-pppoe-server #Roaring Penguin PPPoE Server 服务器

LuCI → Applications → luci-app-samba #网络共享(Samba)

LuCI → Applications → luci-app-samba4 #网络共享(Samba4)

LuCI → Applications → luci-app-sfe #Turbo ACC网络加速(开启Fast Path转发加速)

LuCI → Applications → luci-app-s-s #SS天朝上网(已弃)

LuCI → Applications → luci-app-s-s #SS天朝上网(已弃)

能不能继续对我哭

```
LuCI -> Applications -> luci-app-simulizad #S111而且内守 S111-WL2201 U
LuCI -> Applications -> luci-app-simple-adblock #简单的广告拦截
LuCI -> Applications -> luci-app-simple-softethervirtual #SoftEther virtual服务器 NAT穿透 (已弃)
LuCI -> Applications -> luci-app-splash #Client-Splash是无线MESH网络的一个热点认证系统
LuCI -> Applications -> luci-app-sqm #流量智能队列管理(QoS)
LuCI -> Applications -> luci-app-squid #Squid代理服务器

LuCI -> Applications -> luci-app-S-S R-plus #S-S R天朝上网Plus+
LuCI -> Applications -> luci-app-S-S R-plus -> Include s-s New Versiong #新SS代理
LuCI -> Applications -> luci-app-S-S R-plus -> Include V2ray #V2Ray透明代理
LuCI -> Applications -> luci-app-S-S R-plus -> Include Kcptun #Kcptun代理
LuCI -> Applications -> luci-app-S-S R-plus -> Include s-sR Server #S-S R客户端
LuCI -> Applications -> luci-app-S-S R-plus -> Include s-sR Socks and Tunnel #S-S R代理

LuCI -> Applications -> luci-app-S-S R-pro #S-S R-Pro
LuCI -> Applications -> luci-app-S-S Rserver-python #s-sR Python服务器
LuCI -> Applications -> luci-app-statistics #流量监控工具
LuCI -> Applications -> luci-app-syncdial #多拨虚拟网卡(原macvlan)
LuCI -> Applications -> luci-app-tinyproxy #Tinyproxy是 HTTP(S) 代理服务器
LuCI -> Applications -> luci-app-transmission #BT 下载工具
LuCI -> Applications -> luci-app-travelmate #旅行路由器
LuCI -> Applications -> luci-app-ttyd #网页终端命令行
LuCI -> Applications -> luci-app-udpxy #udpxy做组播服务器
LuCI -> Applications -> luci-app-uhttpd #uHTTPd Web 服务器
LuCI -> Applications -> luci-app-unbound #Unbound DNS 解析器
LuCI -> Applications -> luci-app-upnp #通用即插即用UPnP(端口自动转发)
LuCI -> Applications -> luci-app-usb-printer #USB 打印服务器
LuCI -> Applications -> luci-app-v2ray-pro #V2Ray透明代理
LuCI -> Applications -> luci-app-vmcsd #KMS服务器设置
LuCI -> Applications -> luci-app-vnstat #vnStat网络监控(图表)
LuCI -> Applications -> luci-app-virtualbypass #virtual BypassWebUI 绕过virtual设置
LuCI -> Applications -> luci-app-vsftpd #FTP 服务器
LuCI -> Applications -> luci-app-watchcat #断网检测功能与定时重启
LuCI -> Applications -> luci-app-webadmin #Web 管理页面设置
LuCI -> Applications -> luci-app-webshell #网页命令行终端 (已弃)
LuCI -> Applications -> luci-app-wifischedule #WiFi 计划
LuCI -> Applications -> luci-app-wireguard #virtual**服务器 WireGuard状态
LuCI -> Applications -> luci-app-wireless-regdb #WiFi无线
LuCI -> Applications -> luci-app-wol #WOL 网络唤醒
LuCI -> Applications -> luci-app-wrtbwmon #实时流量监测
LuCI -> Applications -> luci-app-xlnetacc #迅雷快鸟
LuCI -> Applications -> luci-app-zeroTier #ZeroTier内网穿透
```

原文地址: <https://www.vediotalk.com/archives/6561>

=====【以下的文章为系统自动为您推荐】=====

•

【你比重要的连接】 【MST/EDR固件】 【上链接】

能不能继续对我哭

1.编译环境，我是使用了深度Deepin 15.7系统（64位版），也可以使用Ubuntu16.04 LTS 64位版 2.按CTRL+ALT+T进入终端，更新软件列表，下载安装更新。（注意：仅使用非root账户操作，用root用户编译会出错。） sudo.....

编译自己的OPENWRT软路由固件

OPENWRT/LEDE编译教程 1.首先安装编译需要Linux环境，其他Linux也可以，版本较多，实机安装和虚拟机安都可以，按装步奏自行解决。 2.这里以UBUNTU最新18.04为例。按装完成后一般是英文界面要更换中文请在左下角的.....

openwrt编译入门

Openwrt编译准备工作 1.一台笔记本或台式（这是废话） 2.一个ubuntu系统，现在官方已经出到ubuntu18.0.4 3.一台路由器 4.....

常见的软路由去广告原理及避坑指南

互联网广告无处不在，也是很多站点的收入来源，良性的广告环境有利于站点，用户。需要抵制的是恶意，强制观看，隐私窃取类的恶性广告。使用去广告技术的用户大多会问这样问题： 为什么启动了去广告还是可以看到广.....

IPTV 与互联网融合

前言 依赖 需要一个能够刷 OpenWrt 的路由器(需具有数据包镜像和 udpxy 功能/插件)用于抓包和后续的使用，因为其功耗较低且价格比较便宜 Newifi D2 拼多多 100 以下就可以拿下 能够正常播放节目的 IPTV 机顶盒，如.....

本作品采用 知识共享署名-相同方式共享 4.0 国际许可协议 进行许可

打赏

点赞

分享

openwrt 编译 软路由

最后编辑：2019年12月14日

< 上一篇

下一篇 >

能不能继续对我哭



©2017 - 2020 GXNAS博客. ALL RIGHTS RESERVED.

本博客已建站： 1009天19小时28分13秒

网站备案号：桂ICP备18008717号 | 站点地图 SITEMAP

能不能继续对我哭