

第 14 章 统一接口配置 UCI.....	2
14.1 UCI 系统介绍.....	2
14.2 UCI 系统规则.....	2
14.3 UCI 文件语法.....	3
14.4 UCI 命令.....	4

<https://wy-wulian.taobao.com>

第 14 章 统一接口配置 UCI

本章目标

- 了解什么是 UCI 系统
- 掌握 UCI 文件语法、UCI 命令

14.1 UCI 系统介绍

UCI 即统一接口配置(Unified Configuration Interface)的简称,它被应用于 OpenWrt 系统上面。

UCI 是简单、实用、直接了当的。UCI 是 OpenWrt 上面主要的配置用户接口,用来作为主要的系统设置,这些设置通常对设备是至关重要的,例如网络接口、无线设置、远程登陆权限设置等都是通过 UCI 来实现配置的。它是构建一整套系统的必须部分。

此外,大部分第三方软件也被集成到 UCI 中,因此他们在 openwrt 中也能很简单的配置。许多程序有他们自己的配置文件,像 `/etc/network/interfaces`、`/etc/exports`、`/etc/dnsmasq.conf`、`/etc/samba/smb.conf`,并且他们有他们自己的语法,openwrt 不需要改变他们,只需要改 UCI 设置就行,openwrt 开发者已经对许多软件做了 UCI 集成,当然 openwrt 开发者不可能把所有软件都集成进来。因为他们的精力也是有限的。

大部分被集成到 UCI 系统中的软件通过修改 UCI 配置文件,都能很简单的被配置,这些配置主要应用于系统初始化脚本,他们在 `/etc/init.d` 下面,开启被 UCI 系统集成的进程就需要用 UCI 配置文件。比如运行 `/etc/init.d/samba start` 就会用到 `/etc/config/samba` 配置文件。

14.2 UCI 系统规则

Openwrt 中 UCI 配置文件被放在 `/etc/config` 目录下面,每一个配置文件设计到系统的一种配置。你可以用文本编辑器修改这个配置文件,或者用 `uci` 命令修改。当然它也可以用其他 API 接口来修改,比如 `shell`、`lua` 等,而且 web 接口像 `luci`、`webif` 也可以改变它。当配置文件被改变后,必须重启程序才能生效。

下面举一个修改 ip 地址的例子,如果你想把默认的 ip: 192.168.1.1 改为 192.168.2.1,可以通过 `vi` 来编辑配置文件。

```
$ vi /etc/config/network
```

在 `/etc/config/network` 中找到:

```
option ipaddr 192.168.1.1
```

把他改变为

```
option ipaddr 192.168.2.1
```

然后重启网络 `/etc/init.d/network restart` 就会发现系统的 ip 已经变为新 ip 地址了。

14.3 UCI 文件语法

UCI 配置文件通常一个或者更多的 config 语句，一个 config 语句中包括一个或者更多的 option 语句来表示配置内容。下面一个 UCI 配置文件的内容：

```
package 'example'

config 'example' 'test'
    option 'string'      'some value'
    option 'boolean'     '1'
    list 'collection'    'first item'
    list 'collection'    'second item'
```

`config 'example' 'test'` 这条语句是一个 config 语句的开始，表示类型是 example，test 代表名字，其中类型是必不可少的，而且不同的 config 语句不能有相同的类型，它对启动脚本有非常重要的意义，但是名字不是必不可少的，可以没有，也可以有。

`option 'string' 'some value'` 和 `option 'boolean' '1'` 定义了值的内容，这里 string 和 boolean 在语法上面没有差异。在 boolean 中 '0'，'no'，'off'，'disable'，'false' 都代表 0，而 '1'，'yes'，'on'，'true'，'enable' 都代表 1。

用 list 语句可以定义多个值，他们都有相同的名字，在这里为 collection。

其中 option 和 list 对配置文件是非常重要的，每个字段都是用 option 或者 list 来表示，option 代表字段值唯一（后面叫 option 字段），list 代表字段有多个值（后面叫 list 字段），他们是配置文件赋值的重要表示。但是在一个配置文件中，它们不是必须的语法。通常值不需要用引号括起来，只有有空格或者 TAB 建的时候才需要用引号。而且单引号和双引号要一一对应，下面表示合法的语言：

```
option example value
option 'example' value
option example "value"
option "example" 'value'
option 'example' "value"
```

14.4 UCI 命令

用 vi、脚本、lua 等工具修改 uci 配置文件比较麻烦还容易出错，openwrt 提供了 uci 命令工具来操作 uci 配置文件。uci 工具可以获取一个值，设置一个值等功能。在终端下面运行 uci 命令可以查看他的帮助说明。

Usage: uci [<options>] <command> [<arguments>]

Commands:

```
batch
export      [<config>]
import      [<config>]
changes     [<config>]
commit      [<config>]
add          <config> <section-type>
add_list    <config>.<section>.<option>=<string>
show        [<config>[.<section>[.<option>]]]
get         <config>.<section>[.<option>]
set         <config>.<section>[.<option>]=<value>
delete      <config>[.<section>[.<option>]]
rename      <config>.<section>[.<option>]=<name>
revert      <config>[.<section>[.<option>]]
reorder     <config>.<section>=<position>
```

Options:

```
-c <path>  set the search path for config files (default: /etc/config)
-d <str>   set the delimiter for list values in uci show
-f <file>  use <file> as input instead of stdin
-a         apply new configuration after commit
-L         do not load any plugins
-m         when importing, merge data into an existing package
-n         name unnamed sections on export (default)
-N         don't name unnamed sections
-p <path>  add a search path for config change files
-P <path>  add a search path for config change files and use as default
-q         quiet mode (don't print error messages)
-s         force strict mode (stop on parser errors, default)
-S         disable strict mode
-X         do not use extended syntax on 'show'
```

下面的表格详细的介绍了他们的主要用法:

名字	Config 字段	描述
commit	[<config>]	把改变的 uci 配置的改变写入文件系统中, 所有的 "uci set", "uci add", "uci rename" 和 "uci delete" 都是通过 uci commit 写入 flash 中。
batch	-	执行 uci 脚本
export	[<config>]	导出一个配置, 供使用人员阅读。
import	[<config>]	输入一个 uci 语句。
changes	[<config>]	显示在没有运行 uci commit 之前的所有配置文件改变记录。
add	<config> <section-type>	增加一个 config
get	<config>.<section>[.<option>]	得到 option 字段值
set	<config>.<section>[.<option>]=<value>	设置 option 字段值
delete	<config>[.<section>[.<option>]]	删除一个 option 或者 list 字段
add_list	<config>.<section>.<option>=<string>	增加一个存在的 list 字段链表。
rename	<config>.<section>[.<option>]=<name>	给 option 字段重命名。
show	[<config>[.<section>[.<option>]]]	展示一个配置的具体内容

下面我们举例说明常用的一些用法。

获取一个值:

获取 lan 口的 ip 地址: uci get network.lan.ipaddr

```
root@OpenWrt:/# uci get network.lan.ipaddr
192.168.1.1
```

设置一个值:

设置 lan 口的 ip 地址为 192.168.2.1

uci set network.lan.ipaddr=192.168.2.1

完成之后需要要使其生效, 运行 uci commit

```
root@OpenWrt:/# uci set network.lan.ipaddr=192.168.2.1
root@OpenWrt:/# uci commit
```

设置好之后重启 network, /etc/config/network restart, 我们发现网口 ip 已经被改变了。

运行 ifconfig。

```
root@OpenWrt:/# ifconfig
```

```
br-lan    Link encap:Ethernet  HWaddr 00:11:22:33:44:55
          inet          addr:192.168.2.1                Bcast:192.168.2.255
Mask:255.255.255.0
          inet6 addr: fd18:75c3:2be6::1/60 Scope:Global
          inet6 addr: fe80::211:22ff:fe33:4455/64 Scope:Link
UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
RX packets:16 errors:0 dropped:0 overruns:0 frame:0
TX packets:8 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:4904 (4.7 KiB)  TX bytes:976 (976.0 B)
```

已经变为 192.168.2.1 了。

导出显示一个选项的配置，现在我们导出 uhttp 的配置，效果如下所示：

```
root@OpenWrt:/# uci export uhttpd
package uhttpd

config uhttpd 'main'
    list listen_http '0.0.0.0:80'
    list listen_http '[:]:80'
    list listen_https '0.0.0.0:443'
    list listen_https '[:]:443'
    option home '/www'
    option rfc1918_filter '1'
    option max_requests '3'
    option max_connections '100'
    option cert '/etc/uhttpd.crt'
    option key '/etc/uhttpd.key'
    option cgi_prefix '/cgi-bin'
    option script_timeout '60'
    option network_timeout '30'
    option http_keepalive '20'
    option tcp_keepalive '1'
    option ubus_prefix '/ubus'

config cert 'px5g'
    option days '730'
    option bits '1024'
    option country 'DE'
    option state 'Berlin'
```

```
option location 'Berlin'
option commonname 'OpenWrt'
```

用 `show` 可以展示 `uhttp` 的具体内容, 和 `export` 不同的是, `export` 时导出配置文件的内容, `show` 时显示 `uhttp` 配置的值。效果如下:

```
root@OpenWrt:/# uci show uhttpd
uhttpd.main=uhttpd
uhttpd.main.listen_http=0.0.0.0:80 [::]:80
uhttpd.main.listen_https=0.0.0.0:443 [::]:443
uhttpd.main.home=/www
uhttpd.main.rfc1918_filter=1
uhttpd.main.max_requests=3
uhttpd.main.max_connections=100
uhttpd.main.cert=/etc/uhttpd.crt
uhttpd.main.key=/etc/uhttpd.key
uhttpd.main.cgi_prefix=/cgi-bin
uhttpd.main.script_timeout=60
uhttpd.main.network_timeout=30
uhttpd.main.http_keepalive=20
uhttpd.main.tcp_keepalive=1
uhttpd.main.ubus_prefix=/ubus
uhttpd.px5g=cert
uhttpd.px5g.days=730
uhttpd.px5g.bits=1024
uhttpd.px5g.country=DE
uhttpd.px5g.state=Berlin
uhttpd.px5g.location=Berlin
uhttpd.px5g.commonname=OpenWrt
```

前面大部分是 `option` 字段的内容, 现在说一下怎么增加 `list` 字段。首先看一下 `system` 的配置内容, 他的 `ntp` 服务器就是 `list` 字段。用 `export` 我们先看一下:

```
root@OpenWrt:/# uci export system
package system

config system
    option hostname 'OpenWrt'
    option timezone 'UTC'

config timeserver 'ntp'
```

```
list server '0.openwrt.pool.ntp.org'
list server '1.openwrt.pool.ntp.org'
list server '2.openwrt.pool.ntp.org'
list server '3.openwrt.pool.ntp.org'
option enabled '1'
option enable_server '0'
```

看到 `server` 就是 `list` 字段,从表格可以看到,`add_list` 可以增加 `list` 字段。`uci add_list system.ntp.server='4.de.pool.ntp.org'` 这个命令可以增加 `4.de.pool.ntp.org` 进入 `list` 字段。增加完之后可以看一下,如下所示:

```
root@OpenWrt:/# uci export system
package system
```

```
config system
```

```
option hostname 'OpenWrt'
option timezone 'UTC'
```

```
config timeserver 'ntp'
```

```
list server '0.openwrt.pool.ntp.org'
list server '1.openwrt.pool.ntp.org'
list server '2.openwrt.pool.ntp.org'
list server '3.openwrt.pool.ntp.org'
list server '4.de.pool.ntp.org'
option enabled '1'
option enable_server '0'
```

如果增加一个新的配置,只需要按照 UCI 系统的语法,增加一个配置文件就可以,或者用 `uci add` 命令。下面列出用 `uci` 命令增加一个新配置的用法:

```
root@OpenWrt:/# touch /etc/config/playapp
```

新建一个配置文件

```
root@OpenWrt:/# uci show playapp
```

显示配置内容,可以看到无内容

```
root@OpenWrt:/# uci add playapp blah
```

增加一个配置项

```
root@OpenWrt:/# uci show playapp
```

显示配置内容

```
playapp.@blah[0]=blah
```

```
root@OpenWrt:/# uci commit && cat /etc/config/playapp
```

```
config blah
```

显示文件内容,可以看到用 `uci` 命令设置进去了。

其他的命令大家可以都自己试一试，这里就不一一列举了。

Opewnrt 界面采用 `luci`，他很好的应用了 `uci` 系统，学好 `uci` 系统对学习 `luci` 有很好的帮助，所以大家需要好好学习这个章节。

注意：

- 1). 该教程为我司 (<https://wy-wulian.taobao.com/>) 原创教程，版权所有；
- 2). 该教程会不断更新、不断深入，详情请咨询我司客服；
- 3). 针对该教程，我们还有 QQ 群和论坛，专门负责技术答疑，详情请咨询我司客服。