

Legacy RUTOS Software Development Kit instructions

[Main Page](#) > [FAQ](#) > [Other Topics](#) > **Legacy RUTOS Software Development Kit instructions**

A **software development kit (SDK)** is a set of software development tools that provides the possibility to create applications for a certain software package, software framework, computer system or similar development platform.

Note: The information in this page is in accordance to using SDK version RUT9XX_R_00.06.00.4. You may read the README file for additional information if using a different SDK version.

□

Contents

- [1 Introduction](#)
- [2 Prerequisites](#)
- [3 Compiling a standard firmware](#)
- [4 Changing default settings](#)
- [5 Firmware branding](#)

Introduction

This article contains instructions on how to utilize Teltonika SDK packages for RUT2xx, RUT850 and RUT9xx routers. The resources used in the provided examples are:

- A [RUT955](#) router
- [Ubuntu 18.04 OS](#)
- SDK version RUT9XX_R_00.06.00.4

However, the examples apply to other routers, SDK versions and operating systems. In most cases the difference will only be that of a visual nature.

Most examples provided in this page are independent from each other. But it is highly recommended to acquaint yourself with the basics by reading the "[Prerequisites](#)" and "[Compiling a standard firmware](#)" sections first as they contain information that will be necessary in order to understand some aspects of the other examples.

It should also be noted early that the first time you compile a firmware it may take **about two hours** to complete. On later attempts the duration is considerably lower.

Prerequisites

You will need:

- A PC, laptop or virtual machine running Linux OS (preferably Ubuntu distro)

- A router from the RUT2xx, RUT9xx series or RUT850
- An SDK intended for your router, which can be downloaded here: [Software Development Kit](#)

Compiling a standard firmware

First, you must install the packages required for the SDK to work. Open the Terminal application (**Ctrl + T**) and execute the following commands:

```
sudo apt-get update
sudo apt-get install -y build-essential curl devscripts gawk gcc-multilib
gengetopt gettext git groff file flex libncurses5-dev libssl-dev python2.7
subversion unzip vim-common zlib1g-dev wget
```

Create a new folder (for this example I'll be using a directory called *RUT9XX_R_GPL_00.06.02* found at *~/Desktop/*) and extract the SDK archive inside it. You can achieve this with drag and drop or by executing this command via Terminal:

```
tar -xf ~/Downloads/RUT9XX_R_GPL_00.XX.YY.tar -C RUT9XX_R_GPL
```

Note: don't forget to replace the file name and path in accordance with your own circumstances

Open a Terminal inside the SDK directory. You can either change the directory in your current terminal (*cd ~/Desktop/RUT9XX_R_GPL_00.06.02/*) or open the directory, right click any white space and click *Open in Terminal*:



Once you open the Terminal, update the feeds:

```
./scripts/feeds update -a
```

And then you can compile a standard firmware by executing the *make* command. If all is in order, the output should look something like this:



Note: the first time you compile a firmware file it may take up to **two hours** before it is complete. Don't close the Terminal window up until then. Once it is finished, you will find the firmware in the *./bin/ar71xx/tltFws/* directory. It should contain two files; the one with **WEBUI** in its name (for example, *RUT9XX_R_GPL_00.06.00.4_000_WEBUI.bin*) can be used to upgrade your router's firmware via its web interface.

Changing default settings

To create a firmware with different default settings, you must change the default in the config files, which are contained in */RUT9XX_R_GPL_00.06.00.4/package/*. However, there is no unifying system regarding where one should look for config files related to specific services. Therefore, it is very important to acquaint yourself with the **UCI system** (RutOS configuration file system) in order to successfully navigate through the files:

- [Click here](#) for information on the configuration hierarchy
- [Click here](#) to find what configs are related to which services

In many cases you will find that service names as they are displayed in the router's WebUI are similar to the names of the programs responsible for these services. For instance, [VRRP](#) settings can be changed in the `/RUT9XX_R_GPL_00.06.00.4/package/vrrpd/files/` directory. However, if you open the aforementioned directory, you will find that it contains four files:



As you can see from the image above, only one of the files holds the VRRP configuration, while other files contain scripts related to the service. This will be different for each case. For example, among L2TP files, which can be found in `/RUT9XX_R_GPL_00.06.00.4/package/xl2tpd/files`, you will find even more items, including different config files which serve a different purpose:



Firmware branding

If you wish to put a brand on your firmware, you can edit the **config.h** file. The file is located in the `./RUT9XX_R_00.06.00.4/brand/` directory and can be opened with a text editor application.

To brand the firmware, simply open the `config.h` and change any brands as you see fit:

```
char *strings[]={ "", "Teltonika", "www.teltonika.lt",  
"http://www.teltonika.lt/", //0,1,2,3 - default, company name, company web,  
company web link  
"RUT9XX", "RUT900 3G", "RUT905 3G", "RUT950 LTE", "RUT955 LTE", "RUT951  
LTE", "RUT956 LTE", //4-10  
"Teltonika RUT9XX", "Teltonika RUT900 3G", "Teltonika RUT905 3G",  
"Teltonika RUT950 LTE", "Teltonika RUT955 LTE", //11-15  
"Teltonika RUT951 LTE", "Teltonika RUT956 LTE", "Teltonika RUT750",  
"Teltonika RUT755", "Teltonika solutions", //16-20  
"Teltonika_Router", """, "Teltonika", """, """, """, //21-26 - ssid, hostname,  
Wifi page text, routername (get from flash if empty), 950 product code (get  
from flash if empty) , 955 code  
"https://wiki.teltonika-networks.com/index.php?title=Network_products",  
"Wiki Teltonika", // 27-28 Footer link to Wiki  
"https://wiki.teltonika-networks.com/index.php?title=RMS", "Teltonika Wiki  
information" //29-30 - RMS table link to RMS Wiki page  
};
```

The strings highlighted in **pink** are the different brand names that can be found in the WebUI of the router. Highlighted in **blue** are descriptions on where these names may be encountered.

Take note the firmware is intended for use not on a single router but on an entire series of routers (RUT9xx in the case of this example). That is why you see many different router names (RUT900, RUT905, RUT950, etc.) If you're planning on using your firmware on only one router (only RUT955, for example), you can just change the names that are relevant to you.

The empty quotation marks ("") indicate that the value is omitted and should be obtained from the router's flash memory. This may be important to remember because if you leave the quotations marks empty, your branding may be left incomplete. For example, if you do not indicate the

[hostname](#) value (fifth row, second element; left empty by default), it will be taken from the router's flash memory and will probably not coincide with the rest of your branding.

Once you are done with editing, save the changes made to the file and compile a firmware to test whether your branding was successful.