# OpenWrt: LuCI: how to implement limited user access

Asked 11 years, 8 months ago Modified 3 years, 4 months ago Viewed 16k times



Goal: two users root and user. Root can access everything via web-interface, but user should see only some parts of the menus.





One option would be to pass "sysauth" option to every module in question. That is not very practical, because the user would see every menu entry and would get login page for every menu he is not allowed to.





My idea is to figure out who is logged on and then do nothing in the index() function of each restricted module. So far I couldn't find such a function in LuCI API (<a href="http://luci.subsignal.org/api/luci/">http://luci.subsignal.org/api/luci/</a>), that would return a current logged user.

I know how to add additional users in OpenWrt/LuCI (<a href="https://forum.openwrt.org/viewtopic.php?">https://forum.openwrt.org/viewtopic.php?</a>
<a href="pid=163013#p163013">pid=163013#p163013</a>). But it is only a part of the solution.

Any idea, how to achieve my goal?

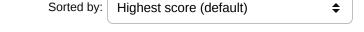


Share Improve this question Follow



#### 2 Answers







I ended up creating a Lua function like described here: <a href="http://lua-users.org/wiki/SaveTableToFile">http://lua-users.org/wiki/SaveTableToFile</a>, to find and remove unneeded keys from the table.

3









```
function remove_idx( tbl, index )

-- initiate variables for save procedure
local tables, lookup = { tbl }, { [tbl] = 1 }

for idx, t in ipairs( tables ) do
    local thandled = {}

    for i, v in ipairs( t ) do
    thandled[i] = true
    local stype = type( v )
    -- only handle value
    if stype == "table" then
        if not lookup[v] then
```

```
table.insert( tables, v )
           lookup[v] = #tables
     else
        if i == index then
           t[i] = nil
           return
        end
     end
      end
      for i, v in pairs(t) do
     -- escape handled values
     if (not thandled[i]) then
        local flag = 0
        local stype = type( i )
        -- handle index
        if stype == "table" then
           if not lookup[i] then
          table.insert( tables,i )
          lookup[i] = #tables
           end
        else
           flag = 1
           if i == index then
          t[i] = nil
          return
           end
        end
        if flag == 1 then
           stype = type(v)
           -- handle value
           if stype == "table" then
          if not lookup[v] then
             table.insert( tables, v )
             lookup[v] = #tables
          end
           else
          if i == index then
             t[i] = nil
             return
          end
           end
        end
     end
      end
   end
end
```

And then inserted my user check and page delete after in libs/web/luasrc/dispatcher.lua dispatch():

```
if c and c.index then
  local tpl = require "luci.template"

if util.copcall(tpl.render, "indexer", {}) then
  return true
```

```
end
end
```

That's how I remove unneeded pages depending on who is logged in:

It is a little bit quick and dirty, but it works. Please note, that direct access by manipulating the URL is still possible.

### **Update**

LuCl2 will provide ACL support und multi-user environment: <a href="http://git.openwrt.org/?">http://git.openwrt.org/?</a>
<a href="psi-project/luci2/ui.git;a%3Dsummary">p=project/luci2/ui.git;a%3Dsummary</a>

Share Improve this answer Follow

edited Jan 20, 2014 at 8:42



thanks for share your solution. i have a simple question: i don't know the remove\_idx write to which file? add "user check" after dispatch() function or in dispatch() 'modifi dispatch()'. and where i use "remove unneeded pages" code? I'm confused. sorry for my bad english. — omid Jan 10, 2014 at 12:22

i added remove\_idx function and user check, but only remove item in menu first page render, if reload or go to other page, leds and packages item go back and available in menu. may be i bad use check user code, can you help me please? — omid Jan 11, 2014 at 20:03

Both remove\_idx() and authentication check must be in libs/web/luasrc/dispatcher.lua file, because the whole index tree is built as soon as you open routers web interface. User authentication check must be in the dispatch() routine, right after the code I showed in my answer. This is the routine, where the index tree will be filled. — yegorich Jan 11, 2014 at 21:13 /

i added if ctx.authuser... after if c and c.index..., if i click system or network menu, leds and packages not available in menu list but if i click dhcp or startup ..., leds and packages available in menu list. — omid Jan 11, 2014 at 21:23

1 I've solved the problem of reappearing sub-tabs through replacing c with ctx.tree in remove\_idx() call. But direct access by manipulating the URL is still possible. – yegorich Jan 20, 2014 at 8:40



If you'd like to create multiple OpenWRT Luci users with varying access, you can following these steps:

2

1. Create a local user account



2. Add the user to the RCP configuration and define access level

See sample excerpt from /etc/config/rpcd config below:



This also works if you're obtaining an authentication token for <u>JSON-RPC</u> calls to Luci.

Share Improve this answer Follow

answered Dec 21, 2021 at 21:44

empiric53
21 • 3

Any idea how to further limit read access e.g. to specific values? - carbolymer Jan 3, 2022 at 12:01

#### Start asking to get answers

Find the answer to your question by asking.

Ask question

## **Explore related questions**

authentication lua openwrt

See similar questions with these tags.