

IPv6 Tunnel on OpenWrt

The objective of this platform is to connect your home network to the Internet using an IPv6 tunnel provided by [sixxs](#). A TP-Link MR3020 router is used as the tunnel endpoint:

- The Ethernet interface is connected to your home set-top box.
- The WiFi interface creates a local area network and connects the home equipment.

Start by configuring Ethernet and WLAN interfaces on the MR3020 in two different IP networks. As the Ethernet interface is connected to the set-top box, it is important to adapt the following configuration to your environment by choosing an appropriate subnet and gateway. The WiFi interface has an address in a subnet of your choice.



Note that in OpenWrt, the configuration uses logical names for interfaces, such as wan for eth0.

[/etc/config/network](#)

```
config interface 'loopback'
    option ifname 'lo'
    option proto 'static'
    option ipaddr '127.0.0.1'
    option netmask '255.0.0.0'

config interface 'wan'
    option ifname 'eth0'
    option proto 'static'
    option ipaddr '192.168.1.66'
    option netmask '255.255.255.0'
    option gateway '192.168.1.1'
    option dns '8.8.8.8'

config interface 'wifi'
    option proto 'static'
    option ipaddr '192.168.66.1'
    option netmask '255.255.255.0'
    option ip6addr '2a01:240:xxxx:yyyy::1/64'
```

[/etc/config/dhcp](#)

```
config dnsmasq
    option domainneeded 1
    option boguspriv 1
    option filterwin2k 0 # enable for dial on demand
    option localise_queries 1
    option rebind_protection 1 # disable if upstream must serve
```

```
RFC1918 addresses
option rebind_localhost 1 # enable for RBL checking and similar
services
#list rebind_domain example.lan # whitelist RFC1918 responses for
domains
option local '/lan/'
option domain 'lan'
option expandhosts 1
option nonegcache 0
option authoritative 1
option readethers 1
option leasefile '/tmp/dhcp.leases'
option resolvfile '/tmp/resolv.conf.auto'
#list server '/mycompany.local/1.2.3.4'
#option nonwildcard 1
#list interface br-lan
#list notinterface lo
#list bogusnxdomain '64.94.110.11'

config dhcp wifi
option interface wifi
option start 100
option limit 150
option leasetime 12h

config dhcp wan
option interface wan
option ignore 1
```

[/etc/config/wireless](#)

```
config wifi-device radio0
option type mac80211
option channel 11
option macaddr f8:d1:11:bd:62:ce
option hwmode 11ng
option htmode HT20
list ht_capab SHORT-GI-20
list ht_capab SHORT-GI-40
list ht_capab RX-STBC1
list ht_capab DSSS_CCK-40

config wifi-iface
option device radio0
option network wifi
option mode ap
option ssid 'PocketBox'
option encryption 'psk2'
option key '1CAFE2DECA'
```

[/etc/config/firewall](#)

```
config zone
    option name            wifi
    option input           ACCEPT
    option output          ACCEPT
    option forward         REJECT

config zone
    option name            lan
    option network         'lan'
    option input           ACCEPT
    option output          ACCEPT
    option forward         REJECT

config zone
    option name            wan
    option network         'wan'
    option input           REJECT
    option output          ACCEPT
    option forward         REJECT
    option masq            1
    option mtu_fix         1

config forwarding
    option src             lan
    option dest            wan

config forwarding
    option src             wifi
    option dest            wan
```

```
opkg update && opkg install aiccu radvd
```

```
ip -6 addr add 2a01:240:fe00:80ab::1/64 dev wlan0
```

[/etc/sysctl.conf](#)

```
net.ipv6.conf.all.forwarding=1
```

[/etc/config/radvd](#)

```
config interface
    option interface      'wifi'
    option AdvSendAdvert  1
    option AdvManagedFlag 0
    option AdvOtherConfigFlag 0
    list client           ''
```

```

option ignore          0

config prefix
option interface      'wifi'
# If not specified, a non-link-local prefix of the interface is
used
list prefix          '2a01:240:fe00:80ab::/64'
option AdvOnLink     1
option AdvAutonomous 1
option AdvRouterAddr 0
option ignore        0

config route
option interface      'wifi'
list prefix          ''
option ignore        1

config rdns
option interface      'wifi'
# If not specified, the link-local address of the interface is used
list addr            ''
option ignore        1

config dns
option interface      'wifi'
list suffix          ''
option ignore        1

```

Create and edit

[/etc/aiccu.conf](#)

```

# AICCU Configuration

# Login information (defaults: none)
username XXXX-SIXXS
password XXXX

# Protocol and server to use for setting up the tunnel (defaults: none)
#protocol <tic|tsp|l2tp>
#server <server to use>

# Interface names to use (default: aiccu)
# ipv6_interface is the name of the interface that will be used as a
tunnel interface.
# On *BSD the ipv6_interface should be set to gifX (eg gif0) for
proto-41 tunnels
# or tunX (eg tun0) for AYIYA tunnels.
ipv6_interface tun0

```

```
# The tunnel_id to use (default: none)
# (only required when there are multiple tunnels in the list)
tunnel_id T23724

# Be verbose? (default: false)
verbose true

# Daemonize? (default: true)
# Set to false if you want to see any output
# When true output goes to syslog
#
# WARNING: never run AICCU from DaemonTools or a similar automated
# 'restart' tool/script. When AICCU does not start, it has a reason
# not to start which it gives on either the stdout or in the (sys)log
# file. The TIC server *will* automatically disable accounts which
# are detected to run in this mode.
#
daemonize true

# Automatic Login and Tunnel activation?
automatic true

# Require TLS?
# When set to true, if TLS is not supported on the server
# the TIC transaction will fail.
# When set to false, it will try a starttls, when that is
# not supported it will continue.
# In any case if AICCU is build with TLS support it will
# try to do a 'starttls' to the TIC server to see if that
# is supported.
requiretls false

# PID File
#pidfile /var/run/aiccu.pid

# Add a default route (default: true)
#defaultroute true

# Script to run after setting up the interfaces (default: none)
#setupscript /usr/local/etc/aiccu-subnets.sh

# Make heartbeats (default true)
# In general you don't want to turn this off
# Of course only applies to AYIYA and heartbeat tunnels not to static
ones
#makebeats true

# Don't configure anything (default: false)
#noconfigure true

# Behind NAT (default: false)
```

```
# Notify the user that a NAT-kind network is detected
behindnat true

# Local IPv4 Override (default: none)
# Overrides the IPv4 parameter received from TIC
# This allows one to configure a NAT into "DMZ" mode and then
# forwarding the proto-41 packets to an internal host.
#
# This is only needed for static proto-41 tunnels!
# AYIYA and heartbeat tunnels don't require this.
#local_ipv4_override
```

```
aiccu start
/etc/init.d/radvd start
/etc/init.d/radvd enable
```

From:
<http://wiki.lahoud.fr/> - **wikiroute**

Permanent link:
http://wiki.lahoud.fr/doku.php?id=ipv6_tunnel_at_home

Last update: **2015/04/04 18:31**

