

Setting up Wireless Internet for a Caravan Park, Vr 1.2

26/06/2008

1 Introduction

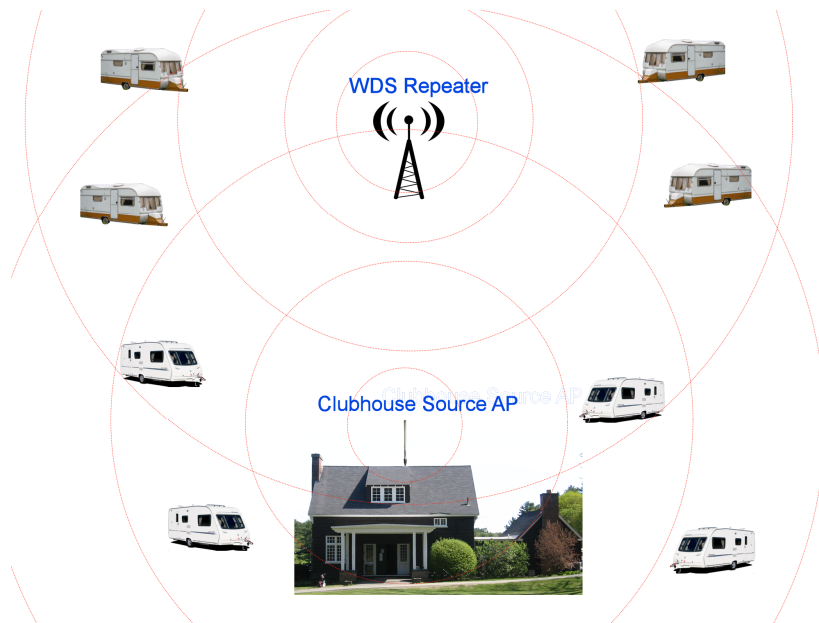
The purpose of this document is to cover some of the design considerations that you need to think about when attempting to layout a wireless setup to deliver internet access to users in a caravan park. However, the topics discussed are equally relevant in any application where the aim is to furnish WiFi internet access to fixed or mobile clients in an outdoor space e.g. you could also use these principles for a marina or park etc...

Frequently, nowadays, caravan site operators want to be able to offer internet access to their customers. The old fashioned method was to simply offer internet access at the main club house however more and more customers are now demanding access in the comfort of their own vans. This is particularly true on static 'van sites where the users could spend several weeks or months staying there – heck some people go into withdrawal if they are away from the internet for a few days let alone months!

The problem with caravan sites are the 'vans: Being mostly metal they form very effective screens for WiFi signals so the aim is to get the WiFi signal to enter the 'van by the windows (usually large ones at each end of the caravan). Generally this means that it's impractical to cover the whole site, giving line of site via the caravan windows, from a single wireless transmission point. We therefore need to have a number of transmission points scattered over the site, arranged to cover all of the vans via their windows.

2 WDS

One way to get a wireless signal to cover the whole site, is to use WDS repeaters carefully placed so that they can link with the nearest WDS station and also give good coverage to the neighbouring 'vans.



Above shows a fairly simple WDS setup.

WDS is quite a good system but it has a big problem related to the way it works. The difficulty is the WDS system doesn't understand routes which means, in order for traffic to get from one part of the site to another, ALL the WDS repeaters in the network wirelessly transmit the data i.e. even if a particular WDS unit isn't anywhere near the send or receiver. Obviously, where you have a high number of WDS units, this creates a huge amount of WiFi congestion. At the very least, since only unit can send or receive wireless data at any one time, then there is a commensurate drop in network performance each time a repeater is added. So, with a single WDS unit, performance is reduce to 50%. With two repeaters it's down to 33% and so on. However this assumes there are no traffic collisions: In practice it's even worse! For this reason you really shouldn't use more than 4 repeaters (in fact most WDS products don't allow more than 4 in the setup) though, personally I wouldn't go for more than 1 or 2 maximum. This limitation on the number of units you can use, coupled with the huge drops in performance really limit a WDS system to small and simple sites. However, as a cheap-n-cheerful, quick to setup solution, for a small site, it's acceptable.

2.1 Equipment

In this case the clubhouse has an Access Point which is fitted with an externally mounted omni antenna. The AP could be an indoor AP with a coax cable running to the outdoor antenna but it would be much better to use a proper outdoor wireless AP which could then be mounted very close to the antenna and thus minimize signal losses in the antenna cable. You then need to connect the WDS AP to a suitable ADSL router (assuming the internet access is via DSL) like the SAR-600ER or, to also give wireless coverage within the club house use a wireless router like the SAR-600EW (note the wireless router needs to operate on a different radio channel so as to not interfere with the main, external, WDS system).

Suitable indoor AP's could, for example, be the excellent EnGenius NCB-3220 or the lower cost BWAP-608(H), or the BIP-W610HP. The outdoor radio device recommended for this application would be the EnGenius EOC-3220EXT – this has excellent power output and signal sensitivity at a realistic price.

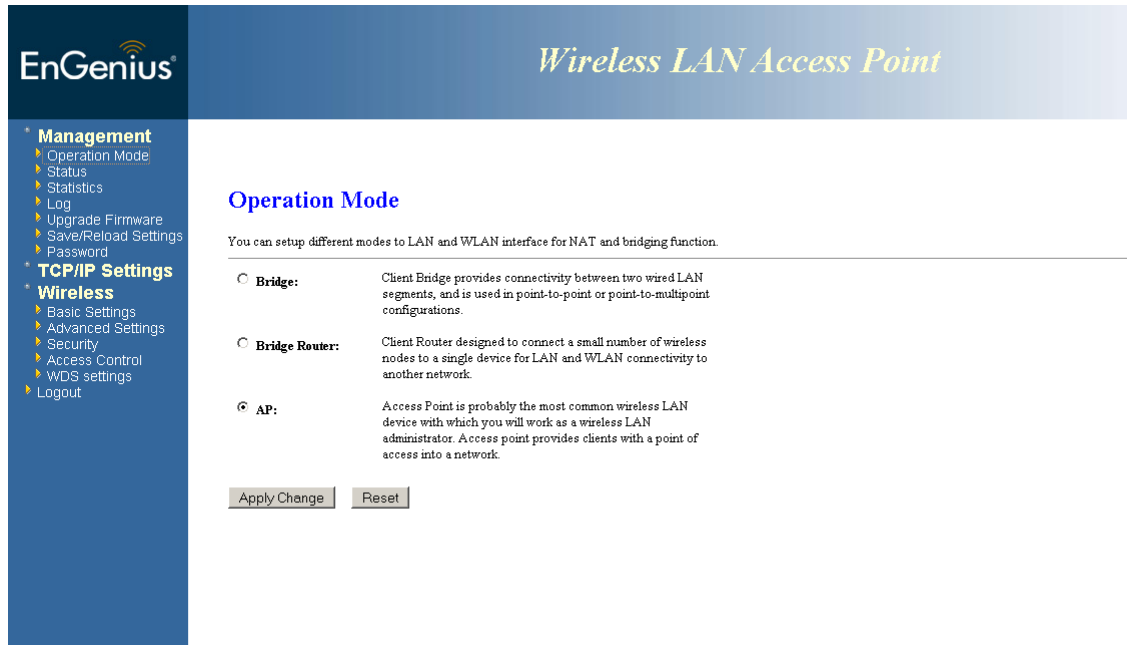
The preferred antenna is a lowish gain omni design which gives a nice, thick beam pattern: There's no point in putting an extremely high gain omni which ends up putting a very high power but thin disc of signal that skims over the roof of

the 'vans and over everyone's heads! You need a signal beam that will reach the ground and give maximum coverage. A suggestion would be an 8dB antenna like the NET-WL-ANT-008ON.

Equipment wise, the WDS Repeater is very similar i.e. it's an AP (indoor if you want but an outdoor unit like the EOC-3220EXT would be better) which can run AP/WDS mode. You also need a low gain omni antenna; again this could be the 8dB WL-ANT-008ON.

2.2 AP Configuration

Assuming you've gone for the EnGenius 3220 product then setup of the AP (main station) is very easy. With the settings as defaults, goto Management/Operation Mode and set AP as the operation mode and then Apply Changes, allow unit to reboot (on the new IP address of 192.168.1.2):



The screenshot shows the EnGenius web interface for a Wireless LAN Access Point. The left sidebar contains a navigation menu with categories: Management (Operation Mode, Status, Statistics, Log, Upgrade Firmware, Save/Reload Settings, Password), TCP/IP Settings, and Wireless (Basic Settings, Advanced Settings, Security, Access Control, WDS settings, Logout). The main content area is titled 'Operation Mode' and includes a sub-header: 'You can setup different modes to LAN and WLAN interface for NAT and bridging function.' Below this, three radio button options are listed: 'Bridge' (described as connecting two wired LAN segments), 'Bridge Router' (designed for connecting wireless nodes to a single device), and 'AP' (selected, described as the most common wireless LAN device). At the bottom of the options are 'Apply Change' and 'Reset' buttons.

For the WDS repeater, also set these to AP mode. Then, after the unit has reset, goto Wireless/WDS Settings. Enable WDS and then add the BSSID of the main station in the WDS AP list (or, if you are repeating from another repeater then enter the BSSID of the WDS/AP that you are repeating from):

- * **Management**
 - ▶ Operation Mode
 - ▶ Status
 - ▶ Statistics
 - ▶ Log
 - ▶ Upgrade Firmware
 - ▶ Save/Reload Settings
 - ▶ Password
- * **TCP/IP Settings**
- * **Wireless**
 - ▶ Basic Settings
 - ▶ Advanced Settings
 - ▶ Security
 - ▶ Access Control
 - ▶ WDS settings
 - ▶ Logout

WDS Settings

Wireless Distribution System uses wireless media to communicate with other APs, like the Ethernet does. To do this, you must set these APs in the same channel and set MAC address of other APs which you want to communicate with in the table and then enable the WDS.

Enable WDS

Add WDS AP: MAC Address Comment

Current WDS AP List:

MAC Address	Comment	Select
00:02:6f:4d:f2:f1	main site	<input type="checkbox"/>

3 Mesh systems



A **wireless mesh network** is a communications network made up of repeating, radio nodes which implement intelligent routing protocols to sensibly route data traffic within the network. The coverage area of the radio nodes, working as a single network, becomes a mesh cloud. Access to this mesh cloud is dependent on the radio nodes working in harmony with each other to create a radio network. A mesh network is reliable and offers redundancy. When one node can no longer operate, all the rest can still communicate with each other, directly or through one or more intermediate nodes. Due to the high degree of redundancy and dynamic routing inherent in the system, wireless mesh networks can self form and self heal and, in multi node configurations, are far superior to WDS or any other repeating topology. They are therefore ideal for distributed wireless access systems e.g. public internet access where you need to provide internet access to large areas and users to caravan sites, etc.

Wireless mesh builds routes between nodes only as desired by originating nodes. It maintains these routes as long as they are needed by the originating node. Wireless mesh nodes forms paths in term of hops which connect together to form the wireless mesh network. Hops are the number of nodes between two a receiving and transmitting client i.e. Laptop, PC, Wi-Fi telephone, IP appliance, etc. Symbolically a Wireless Mesh network is represented by a network cloud.

Mesh nodes uses sequence numbers to ensure the freshness of routes. It is loop-free, self-starting, and scales to large numbers of nodes. Wireless Mesh Nodes builds routes using a route request, route reply query cycle. When a node desires a route to a destination for which it does not already have a route, it broadcasts a route request packet across the network. Nodes receiving this packet update their information for the source node and set up backwards pointers to the source node in the route tables. The wireless nodes also collect other active nodes including IP address, current sequence number, and broadcast ID, and contains the most recent sequence number for the destination of which the source node is aware. A node receiving a route request may send a route reply when it is either the destination or if it has a route to the destination with corresponding sequence number greater than or equal to that contained in the route request. Nodes keep track of the route request through source IP address and broadcast ID. The nodes know when they receive a route request which they have already processed; they discard it and will not forward it.

As the backward pointers propagates back to the originating node, it then sets up forward pointers to the destination. Once the source node receives the backward pointers, it may begin to forward data packets to the destination. When the source later receives a backwards pointer containing a greater sequence number or contains the same sequence number with a smaller hop count, it may update its routing information for that destination and begin using the better route.

As long as the route remains active, it will continue to be maintained. A route is considered active as long as there are data packets periodically travelling from the source to the destination along that path. Once the source stops sending data packets, the links will time out and eventually be deleted from the intermediate node routing tables. Most wireless mesh nodes maintain routes for as long as the route is active. This includes maintaining hops for the life of the cloud. Because the network nodes can be mobile or shut down, it is likely that many link breakages along a route will occur during the lifetime of that route.

3.1 Network Structure

Wireless mesh architecture is a first step towards providing high-bandwidth network over a specific coverage area. Wireless mesh architecture's infrastructure is, in effect, a router network minus the cabling between nodes. It's built of peer-to-peer radio devices that don't have to be cabled to a wired port like traditional WLAN access points (AP) do. Mesh architecture sustains signal strength by breaking long distances into a series of shorter hops. Intermediate nodes not only boost the signal, but cooperatively make forwarding decisions based on their knowledge of the network. Such architecture provides high bandwidth, spectral efficiency, and economic advantage over the coverage area.

Wireless mesh network have a topology inherently more stable than a traditional WDS or universal repeating network. The traffic, being aggregated from a large number of end users, changes infrequently. Practically all the traffic is either

forwarded to or from a gateway, while in ad hoc networks the traffic flows between arbitrary pairs of nodes. A multi hop based nodes proactive routing scheme is used for traffic forwarding, since it easily allows flows aggregation and would minimize overhead, ensuring an optimal utilization of bandwidth.

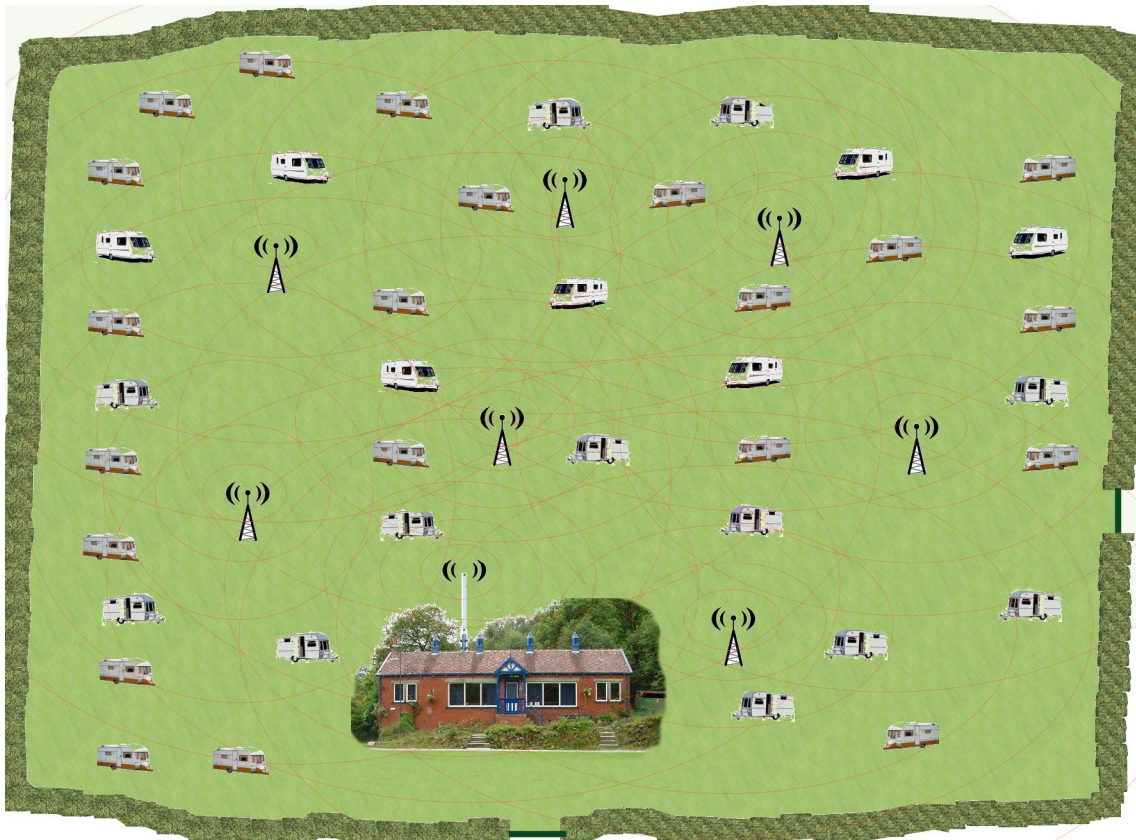
This type of infrastructure can be relatively inexpensive, and very reliable and resilient, as each node needs only transmit as far as the next node. Nodes act as repeaters to transmit data from nearby nodes to peers that are too far away to reach, resulting in a network that can span large distances, especially over rough or difficult terrain. Mesh networks are also extremely reliable, as each node is connected to several other nodes. If one node drops out of the network, due to hardware failure or any other reason, its neighbours simply find another route. Extra capacity can be installed by simply adding more nodes.

3.2 Caravan Site Example

Let's go back to our caravan site example. Previously we discussed using a WDS repeater setup in order to distribute wireless internet over the site. However, there are many advantages to, instead, setting up a site wide meshing wireless network. Going to a meshing setup will allow us to run a higher density of nodes without compromising performance and, gives a more resilient network with higher stability and better accessibility.

3.2.1 Outline

See the figure below (apologies for the rather 'naïve' drawing style ☺):



The illustration depicts a 'typical' 'van site' where a number of 'vans exist where they want to receive wireless internet.



Wireless internet network originates at the 'club house'.



It's then 'repeated' over the site using a series of meshing nodes.

We intentionally use a high density of meshing nodes so that each 'van potentially has more than one node it can use for signal source and each node can see more than one other meshing node: This is something that would just not be possible using a WDS or 'Universal' repeater configuration because the massive resulting drop in performance would just make the whole network unusable! However, by using a high density of nodes then we significantly improve the likelihood that an end user client (someone sat in their 'van, using their wireless notebook to check an eBay bid for some Mickey mouse sunglasses) can get a good wireless connection. Also, multiple overlapping of the nodes means that there are multiple potential routes from source to destination which gives improved link stability: If any node goes out of action of any wireless path becomes blocked, then an alternative route for the traffic will be found and used.

3.2.2 Equipment

3.2.2.1 Software

There are a number of meshing protocols available but one of the more popular is the olsrd system...

<http://www.olsr.org/>

OLSR is a routing protocol for fixed point and mobile ad-hoc networks. The protocol is pro-active, table driven and utilizes a technique called *multipoint relaying* for message flooding.

We've found, the simplest way to implement the olsrd daemon is as part of the APRouter Pro firmware.

http://www.aprouter.com.br/new2006/index2_ing.php

APRouter Pro is AP and Router AP firmware with a very high number of features and includes the olsr mesh daemon. This firmware has been ported onto a number of products using the RTL8186 chipset. These include the EnGenius 3220 products (outdoor version only since the indoor doesn't have adequate amounts of ROM/RAM to operate the Router Pro software), and also the indoor BIP-W610HP product: For the 3220 product the 3rd party APRouter Pro f/w must be loaded in replacement for the standard firmware. The BIP-W610HP indoor AP comes as standard with ver. 6.1 APRouter firmware.

Once the olsr software is correctly configured then it should link all the nodes together as one, complete, peer-to-peer network. In this way the olsr nodes act as a meshing backbone for your IP traffic. However, the olsr nodes will only talk wirelessly to other olsr nodes; the olsr mesh is acting as your data backbone. If you want to offer local wifi client connectivity (e.g. for a wireless notebook) then need to connect a separate Access Point to the LAN port of each olsr node. So, in essence, you have TWO radio devices at each node point: One acting as the olsr mesh node and the second as an AP for local wireless connectivity.

Your wireless clients can then connect to the access points using standard 'infrastructure' mode. If your access points support IAPP (IEEE 802.11F) then the access points will use the olsr mesh as their networking backbone and, when all set on the same wireless configuration, allow your clients to dynamically roam from AP/node to AP/node. It also means, where a client can see more than one AP/node, then the client should talk to that AP which has the stronger signal and, if for any reason that node breaks (wireless link is blocked, for example) then the traffic will still flow via a substitute unit.

In addition connectivity to each node can also be done via the LAN port of any olsr node which means that you can easily cater for wired and also wireless clients.



3.2.2.2 Hardware

As far as radio equipment is concerned then all the nodes are essentially the same equipment i.e. a meshing radio unit with a low gain omni antenna (designed to give maximum coverage rather than outright distance).

e.g. you could use the outdoor version of the EnGenius 3220 product which will take a separate antenna:

<http://www.solwise.co.uk/wireless-outdoor-bridging-noc-3220.htm>

[solwise.co.uk](http://www.solwise.co.uk), sales@solwise.co.uk

This is a 'proper' outdoor rated product which gets its power via the LAN cable; so you don't need to provide a separate power supply.

You can then use an 8dB outdoor, omni antenna to go with this....



You can also use an outdoor 3220 with 8dB omni as an access point for each node.

3.3 OLSR Software Configuration

To setup olsrd using APRouter Pro

This is with ver6.1 APRouter Pro f/w and assumes everything is defaults to start off with.

Assuming a basic, two unit, mesh...

Node1 connected to internet (via any of the LAN ports). In this example the main LAN is subnet 192.168.0.0/255.255.255.0 with our internet router on 192.168.0.11.

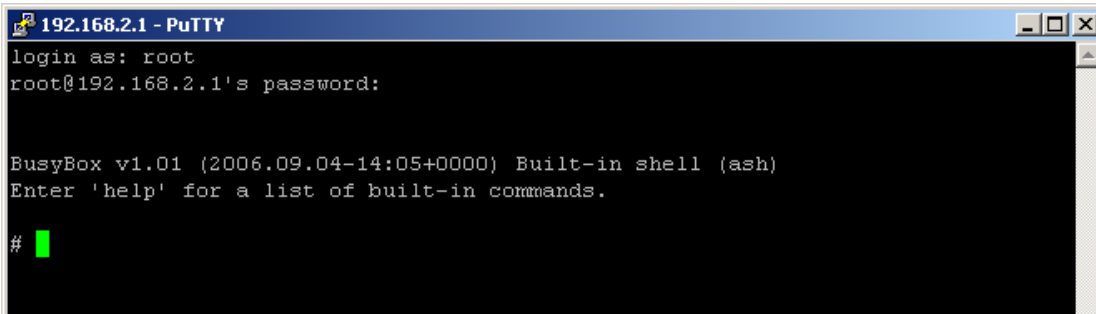
Node2 acting as repeating mesh node

The basics are:

- Make sure that olsrd.conf config file for the main for main node (Node1 – the unit connected to the internet) has HNA4 entry showing that it has internet access (0.0.0.0/0.0.0.0). HNA4 for other nodes should be empty.
- Make sure that olsrd.conf config file for all units has interface “bro” and “wlan0” entry.
- wireless mode is Client/Ad Hoc.
- Set TCP mode as Bridge.
- Give each node a different LAN address (easiest way is to set the LAN ports as DHCP clients so each one will get an address from our main router).
- WAN settings don't matter but you will need to enable Meshing (olsrd) on WAN setup page.
- Add ifconfig wlan0 entry into main script so it's executed each time there's reboot
- Setup Watchdog to reboot if the connection fails for any reason.
- Test...

3.3.1 Configuring the olsrd.conf configuration file

First of all edit the olsrd.conf file. To do that you need to ssh into the unit using PuTTY or similar. Use the username 'root' and password 'admin'

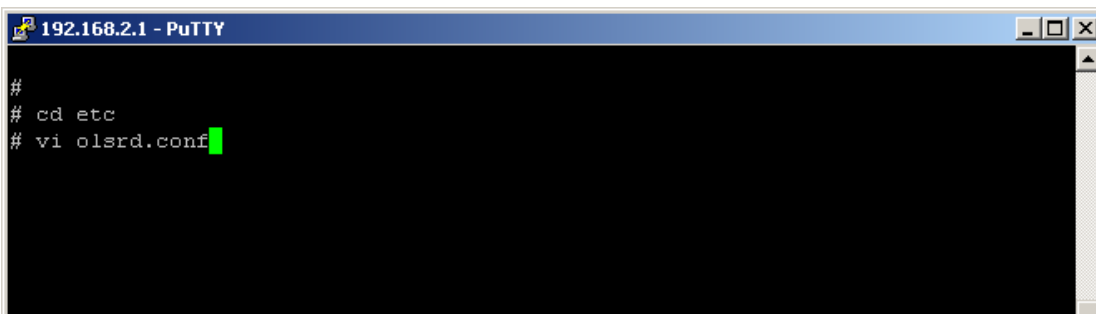


```
192.168.2.1 - PuTTY
login as: root
root@192.168.2.1's password:

BusyBox v1.01 (2006.09.04-14:05+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

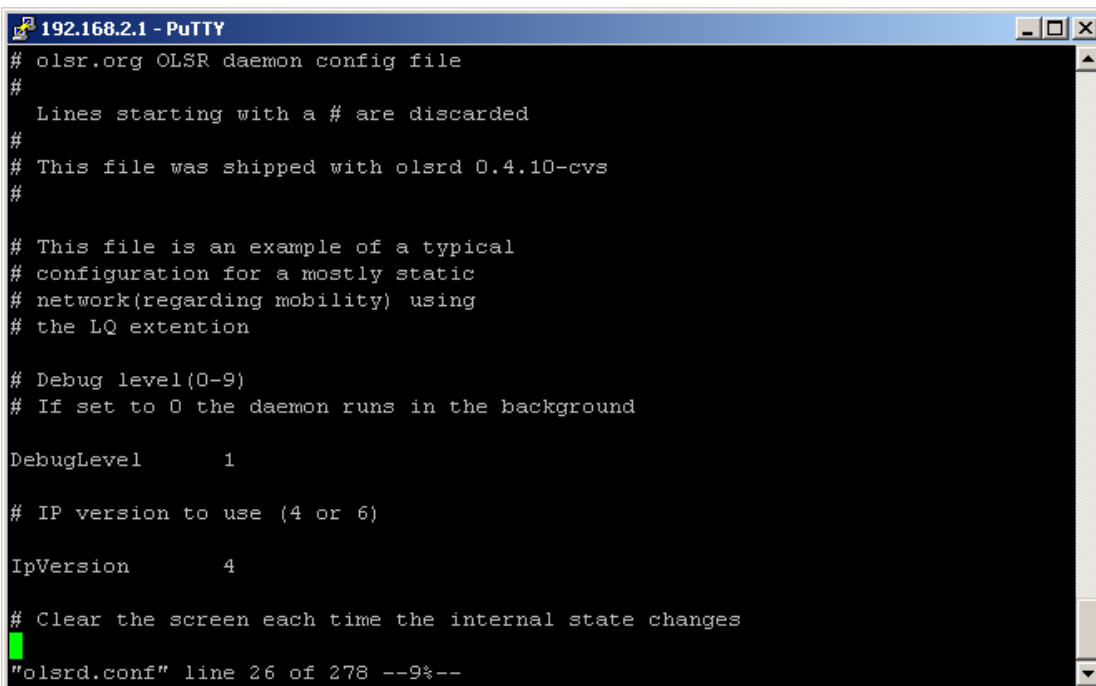
# █
```

Now goto the etc directory and start the vi editor (apologies for this... the vi editor has got to be the worst line editor I've ever, EVER, seen but that's all there is I'm afraid)



```
192.168.2.1 - PuTTY

#
# cd etc
# vi olsrd.conf █
```



```
192.168.2.1 - PuTTY

# olsr.org OLSR daemon config file
#
# Lines starting with a # are discarded
#
# This file was shipped with olsrd 0.4.10-cvs
#
# This file is an example of a typical
# configuration for a mostly static
# network(regarding mobility) using
# the LQ extention

# Debug level(0-9)
# If set to 0 the daemon runs in the background

DebugLevel      1

# IP version to use (4 or 6)

IpVersion       4

# Clear the screen each time the internal state changes
█
"olsrd.conf" line 26 of 278 --9%--
```

Scroll down to the 'Hna4' entry...

```

Hna4
{
#   Internet gateway:
   0.0.0.0  0.0.0.0
#   more entries can be added:
#   192.168.0.0  255.255.255.0
}

# HNA IPv6 routes
# syntax: netaddr prefix
# Example Internet gateway:

```

Now, for Node1, you need to add an entry which tells the olsr network that this is where the main internet access is (0.0.0.0 0.0.0.0). Quick instructions for vi: move cursor to where you want to add. Then do ESCAPE 'i' – this puts you into insert mode. Now when you type then, hopefully, text should appear on the screen (though I agree that vi is REALLY cr*p!). If you do ESCAPE 'x' then it goes into delete mode and, each time you press 'x' it deletes the character under the cursor. However, frequently the display gets mucked up so it's worth doing a few Page Downs and Ups from time to time to make sure it's going according to plan ☺ Anyway... hopefully you can manage to add a line for your network!

For Node2 all entries in the Hna4 section should be remmed out (with '#' at the start of the line).

Now scroll down to the 'Interface' section....

```

192.168.2.1 - PuTTY
# default values. Multiple interfaces
# can be specified in the same block
# and multiple blocks can be set.

# !!CHANGE THE INTERFACE LABEL(S) TO MATCH YOUR INTERFACE(S)!!
# (eg. wlan0 or eth1):

Interface "br0"
{
# IPv4 broadcast address to use. The
# one usefull example would be 255.255.255.255
# If not defined the broadcastaddress
# every card is configured with is used

# Ip4Broadcast          255.255.255.255

# IPv6 address scope to use.
# Must be 'site-local' or 'global'

# Ip6AddrType          site-local

# IPv6 multicast address to use when
"olsrd.conf" line 223 of 278 --80%--

```

The aim here is to change the Interface entry so it just has the br0 an wlan0 interfaces there. So use the ESCAPE 'i' and the ESCAPE 'x' commands to edit the line (I know it's hard but persevere!). Set the same for both Node1 and Node2.

```

# default values. Multiple interfaces
# can be specified in the same block
# and multiple blocks can be set.

# !!CHANGE THE INTERFACE LABEL(S) TO MATCH YOUR INTERFACE(S)!!
# (eg. wlan0 or eth1):

Interface "wlan0" "br0"
(
    # IPv4 broadcast address to use. The
    # one usefull example would be 255.255.255.255
    # If not defined the broadcastaddress
    # every card is configured with is used

    Ip4Broadcast          255.255.255.255

    # Emission intervals.
    # If not defined, RFC proposed values will
    # be used in most cases.

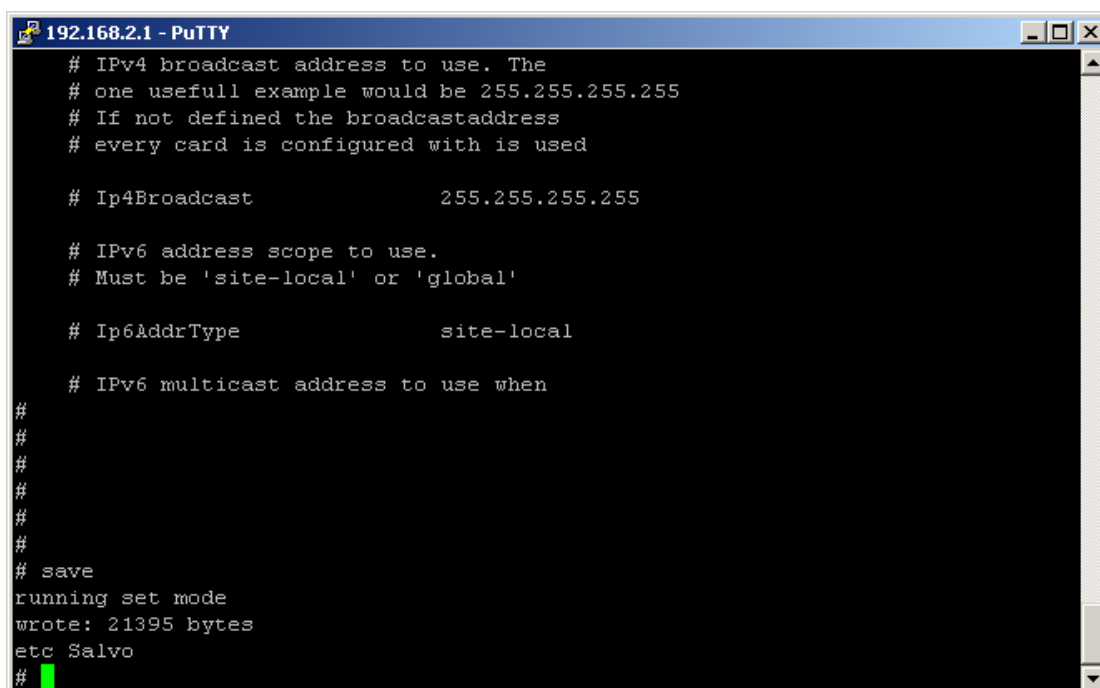
    # Hello interval in seconds(float)
"olsrd.conf" line 167 of 202 --82%--

```

Now we need to write the file and quite the editor.

Do ESCAPE ':w' to write the file. The ESCAPE ':q' to quit vi.

Now you need to ensure the new file is saved to flash so enter the command 'save'



```

192.168.2.1 - PuTTY
# IPv4 broadcast address to use. The
# one usefull example would be 255.255.255.255
# If not defined the broadcastaddress
# every card is configured with is used

# Ip4Broadcast          255.255.255.255

# IPv6 address scope to use.
# Must be 'site-local' or 'global'

# Ip6AddrType          site-local

# IPv6 multicast address to use when
#
#
#
#
# save
running set mode
wrote: 21395 bytes
etc Salvo
#

```

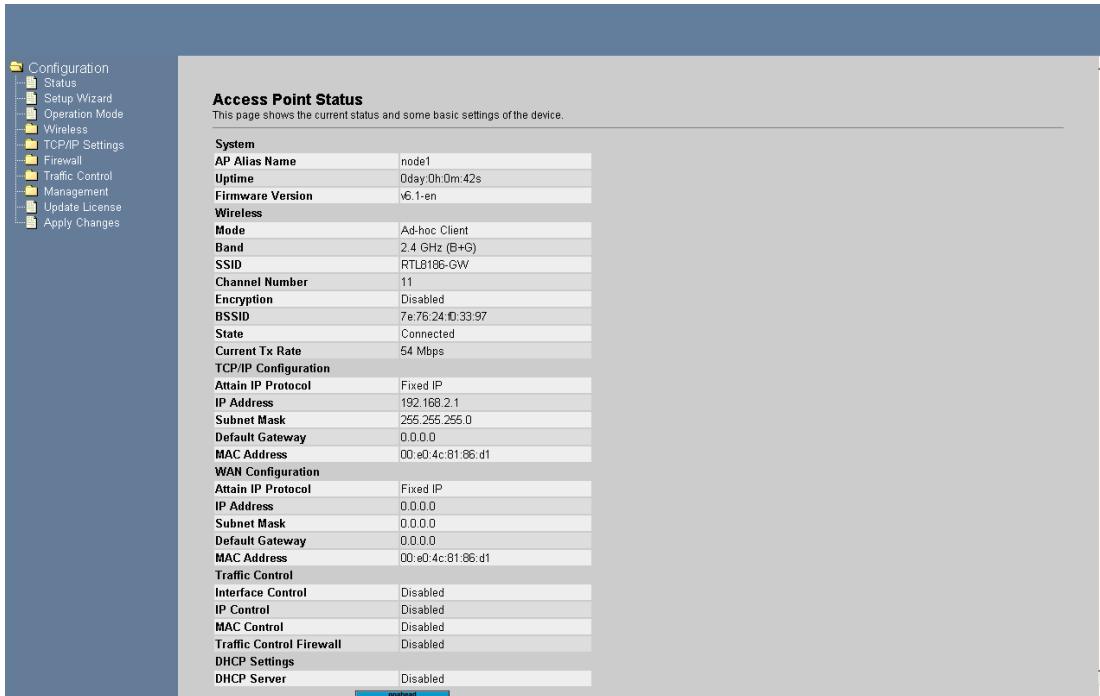
You can now exit PuTTY.

The full olsrd.conf file is listed at the end of the document.

3.3.2 Configuration of the nodes via the web interface

Remember, after doing a change on each web page, you must SAVE the changes.

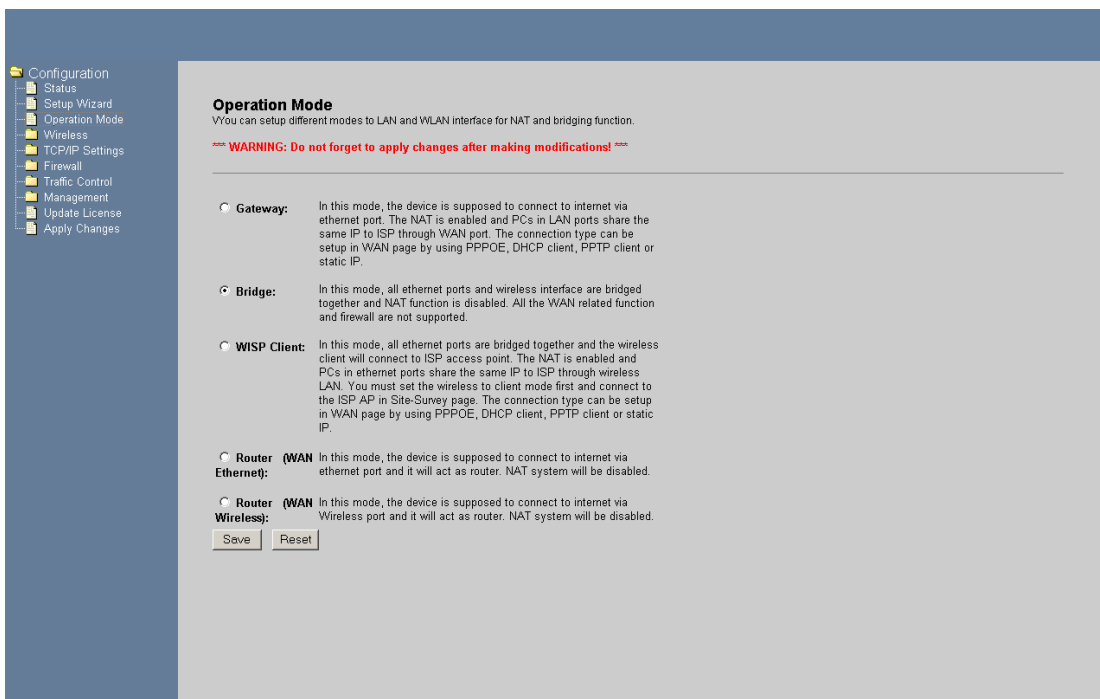
Enter the web setup via your browser (default IP 192.168.2.1)



Access Point Status
This page shows the current status and some basic settings of the device.

System	
AP Alias Name	node1
Uptime	0day 0h 0m 42s
Firmware Version	v6.1-en
Wireless	
Mode	Ad-hoc Client
Band	2.4 GHz (B+G)
SSID	RTL8186-GW
Channel Number	11
Encryption	Disabled
BSSID	7e:76:24:0:33:97
State	Connected
Current Tx Rate	54 Mbps
TCP/IP Configuration	
Attain IP Protocol	Fixed IP
IP Address	192.168.2.1
Subnet Mask	255.255.255.0
Default Gateway	0.0.0.0
MAC Address	00:e0:4c:81:86:d1
WAN Configuration	
Attain IP Protocol	Fixed IP
IP Address	0.0.0.0
Subnet Mask	0.0.0.0
Default Gateway	0.0.0.0
MAC Address	00:e0:4c:81:86:d1
Traffic Control	
Interface Control	Disabled
IP Control	Disabled
MAC Control	Disabled
Traffic Control Firewall	Disabled
DHCP Settings	
DHCP Server	Disabled

Goto Operation Mode – make sure it's Bridge



Operation Mode
You can setup different modes to LAN and WLAN interface for NAT and bridging function.

***** WARNING: Do not forget to apply changes after making modifications! *****

- Gateway:** In this mode, the device is supposed to connect to internet via ethernet port. The NAT is enabled and PCs in LAN ports share the same IP to ISP through WAN port. The connection type can be setup in WAN page by using PPPoE, DHCP client, PPTP client or static IP.
- Bridge:** In this mode, all ethernet ports and wireless interface are bridged together and NAT function is disabled. All the WAN related function and firewall are not supported.
- WISP Client:** In this mode, all ethernet ports are bridged together and the wireless client will connect to ISP access point. The NAT is enabled and PCs in ethernet ports share the same IP to ISP through wireless LAN. You must set the wireless to client mode first and connect to the ISP AP in Site-Survey page. The connection type can be setup in WAN page by using PPPoE, DHCP client, PPTP client or static IP.
- Router (WAN Ethernet):** In this mode, the device is supposed to connect to internet via ethernet port and it will act as router. NAT system will be disabled.
- Router (WAN Wireless):** In this mode, the device is supposed to connect to internet via Wireless port and it will act as router. NAT system will be disabled.

Goto Wireless/Basic – Give unit a name (e.g. Node1 for the main unit, connected to the internet. Node2 for the one acting as a repeating node) and then make sure setup for Client/Ad hoc

Configuration

- Status
- Setup Wizard
- Operation Mode
- Wireless**
 - Basic Settings
 - Advanced Settings
 - Security
 - Access Control
 - WDS settings
 - Site Survey
 - Signal
 - TCP/IP Settings
 - Firewall
 - Traffic Control
 - Management
 - Update License
 - Apply Changes

Wireless Basic Settings

This page is used to configure the parameters for wireless LAN clients which may connect to your Access Point. Here you may change wireless encryption settings as well as wireless network parameters.

***** WARNING: Do not forget to apply changes after making modifications! *****

Disable Wireless LAN Interface

Alias Name:

Band:

Mode:

Network Type:

SSID:

Channel Number:

Reg Domain (Channels):

Associated Clients:

Enable Mac Clone (Single Ethernet Client)

Goto TCP/IP Settings/LAN Interface – give each node a unique address. You can either give each node and address from your main subnet or set the LAN interface as a DHCP client (though, if you do this, you’ll have to check with your DHCP server what address has been allocated)

Configuration

- Status
- Setup Wizard
- Operation Mode
- Wireless
- TCP/IP Settings**
 - LAN Interface
 - WAN Interface
 - IP Aliases
- Firewall
- Traffic Control
- Management
- Update License
- Apply Changes

LAN Interface Setup

This page is used to configure the parameters for local area network which connects to the LAN port of your Access Point. Here you may change the setting for IP address, subnet mask, DHCP, etc.

***** WARNING: Do not forget to apply changes after making modifications! *****

IP Address:

Subnet Mask:

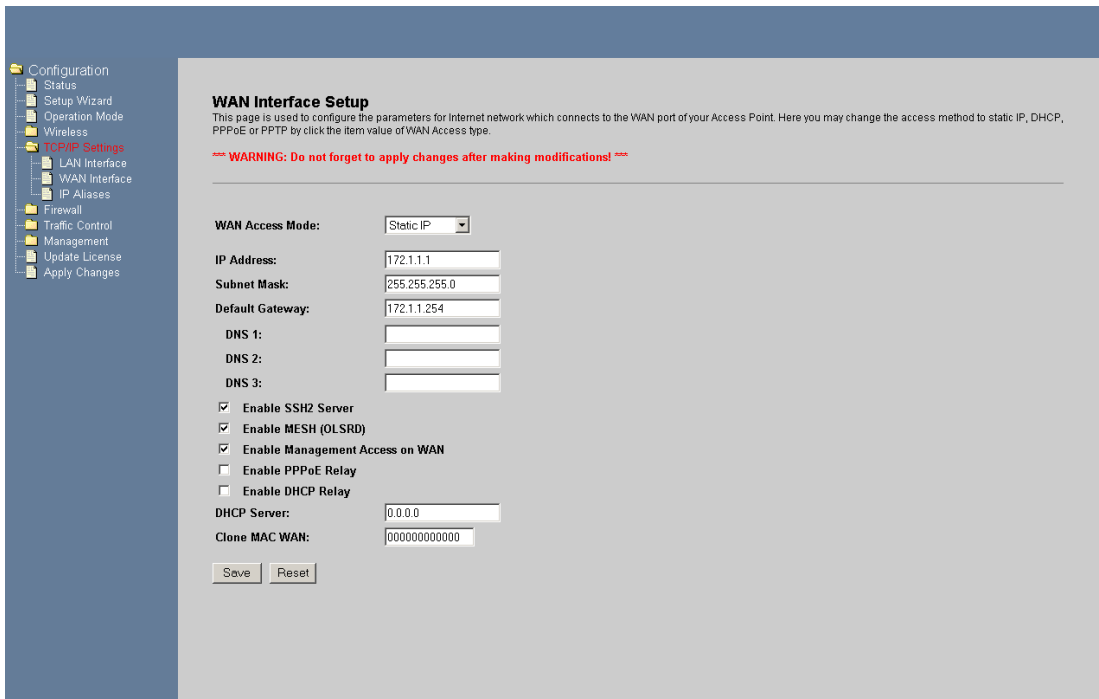
Default Gateway:

DHCP:

DHCP Client Range: -

802.1d Spanning Tree:

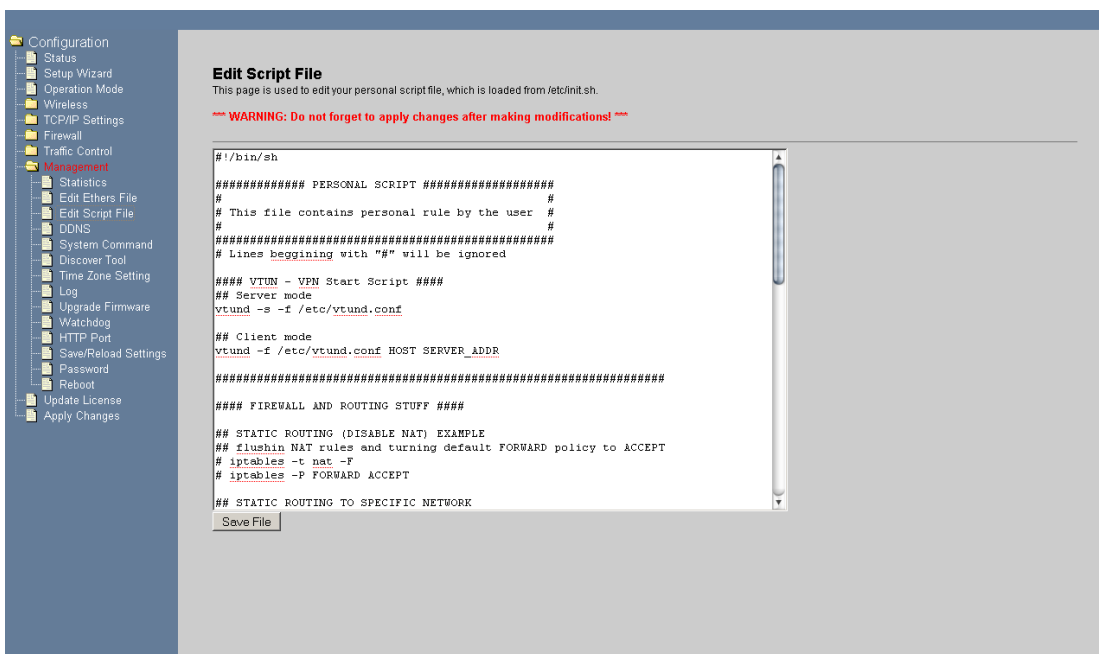
Goto WAN Interface



The WAN interface is not actually used so the IP settings etc... are unimportant. The only important thing is to make sure the box 'Enable MESH (OLSRD)' is ticked.

Now, before the wlan0 interface will load, you need to give it an IP address (olsr is a routing protocol so that means the interfaces need IP addresses).

To do this you can add a simple line to the main startup script called init.sh in the etc directory. The easiest way to do this is via Web gui on the Management/Edit Script File page:

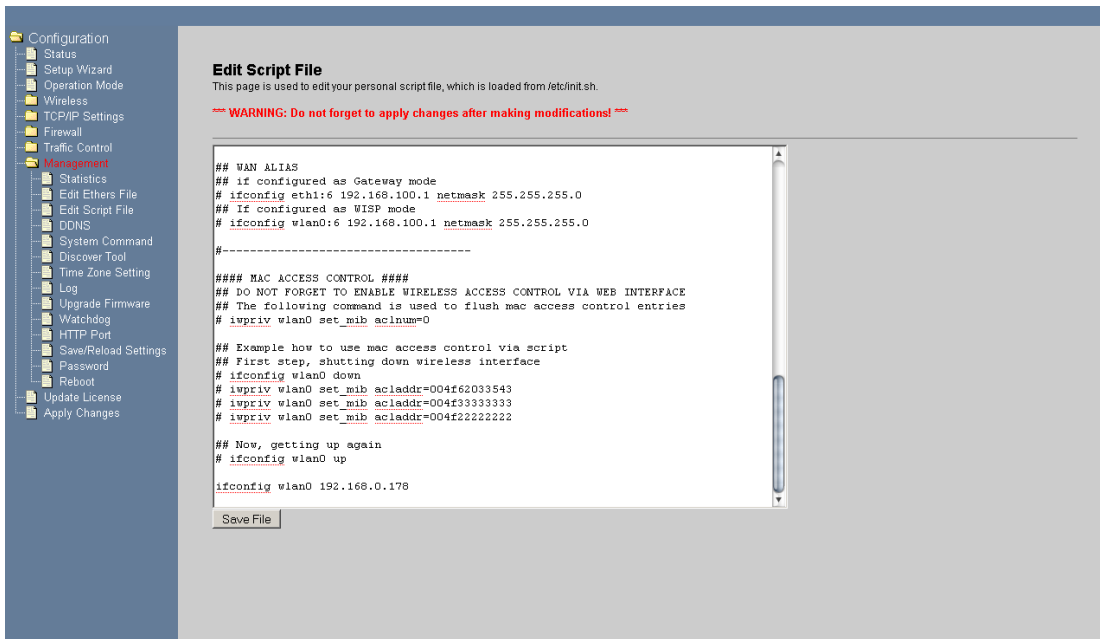


Scroll down to the bottom of the script and add an extra line of the form...

```
ifconfig wlan0 <ip address>
```

Use an unused IP address which is in your main range but outside those used by your DHCP server (e.g. our server only gives out addresses up to 100 so any address above there is free for static use – check with your system if you are unsure!):

e.g.

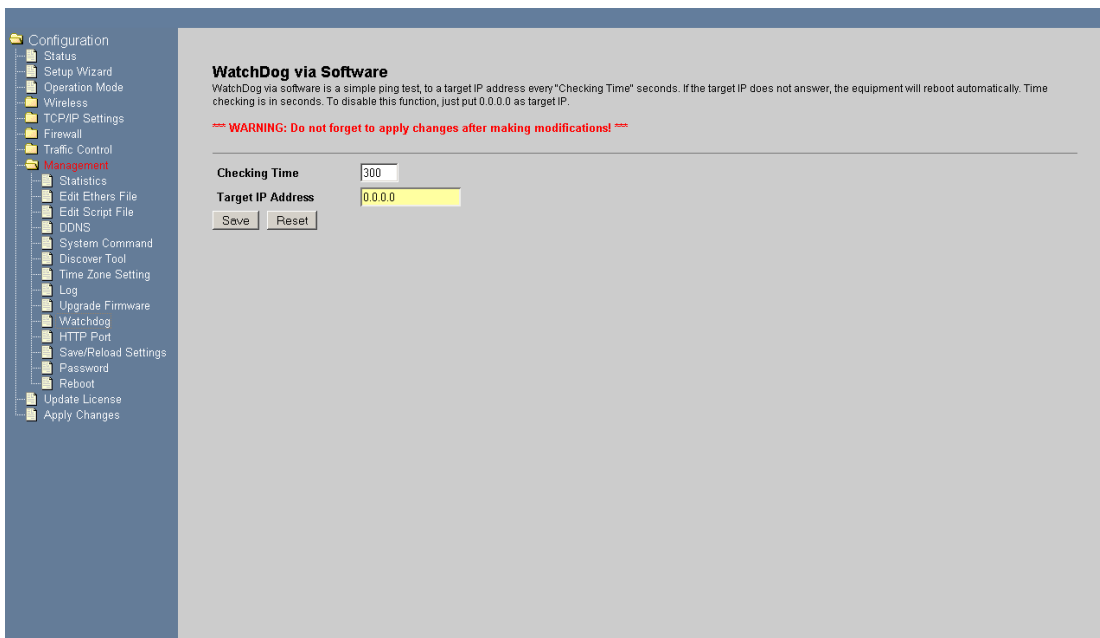


For each node use a different IP address (but still in your main range).

Remember to Save File.

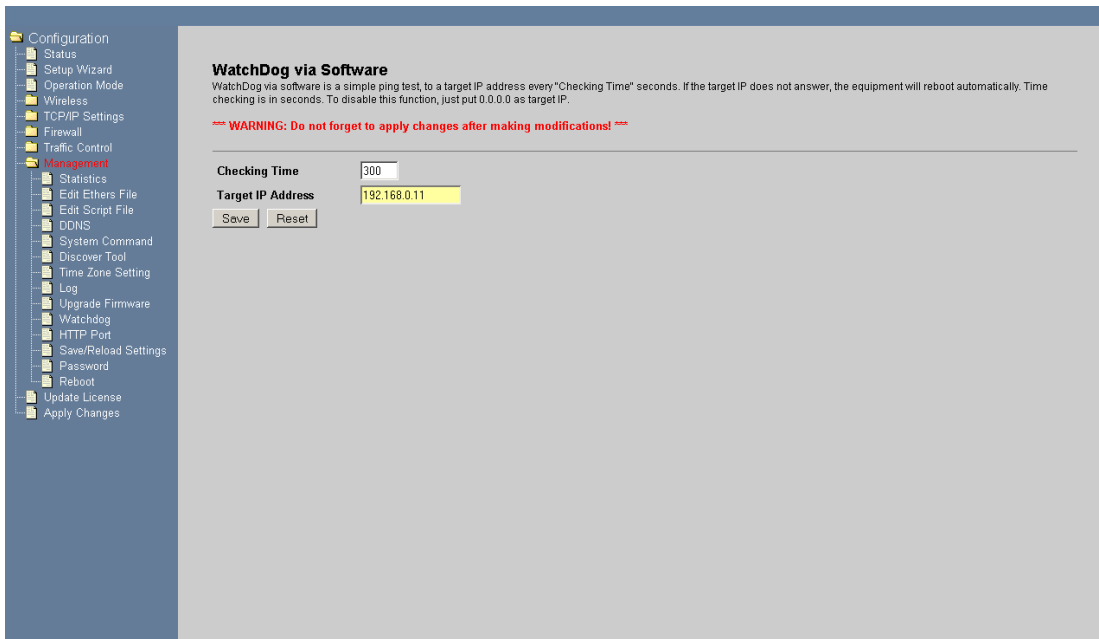
Also it's a good idea to setup the Watchdog (this will reboot the unit if ever it sees something has gone wrong with the link).

To do that goto Management/Watchdog...



You need to enter an IP address for the unit to check connectivity to. I'd suggest the address of your router (on your main LAN). Also set the Checking Time to a sensible value (not too low so that after the slightest hiccup it starts rebooting):

e.g. our router is 192.168.0.11 but use your OWN router address



Remember to save settings and then finally Apply Changes.

Now.... a word of warning, once you enable MESH mode and you have enabled debug mode in the olsrd.conf file then, when you reboot the unit, the web interface is disabled!! Below are notes of how to disable the olsrd via ssh/commands.

3.3.3 Testing

3.4 Basic connectivity – Does it work?

Make sure you've saved all the settings and then reboot the units. Now, if everything's up and running then the obvious test is to see if your wireless PC will connect to the mesh network. When it's connected then, if your PC is set to get IP settings automatically (using DHCP) then your PC should obtain a valid set of IP values and then be able to get onto the internet.

3.5 Checking OLSRD via the meshing units

It's possible to run the meshing in debug mode on the AP units which will log the mesh connectivity. To do this use PuTTY to enter command mode. Then, enter the command 'olsrd'

The olsrd will execute and print out a list of the interfaces...

```

192.168.0.22 - PuTTY
Checking wlan0:
  Wireless interface detected
  Metric: 1
  MTU - IPhdr: 1472
  Index 0
  Address:192.168.0.179
  Netmask:255.255.255.0
  Broadcast address:255.255.255.255
New main address: 192.168.0.179
Checking br0:
  Not a wireless interface
  Metric: 0
  MTU - IPhdr: 1472
  Index 1
  Address:192.168.0.22
  Netmask:255.255.255.0
  Broadcast address:255.255.255.255
Loading plugins...

Main address: 192.168.0.179

Scheduler started - polling every 0.05 seconds
█

```

Below shows the output from a few seconds of running with debug turned on in the olsrd.conf file...

--- 00:59:23.81 ----- LINKS

IP address	hyst	LQ	lost	total	NLQ	ETX
192.168.0.178	0.000	0.200	0	2	1.000	5.00
192.168.0.24	0.000	0.200	0	2	1.000	5.00

--- 00:59:23.81 ----- NEIGHBORS

IP address	LQ	NLQ	SYM	MPR	MPRS	will
192.168.0.178	0.200	1.000	YES	NO	NO	3

--- 00:59:23.81 ----- TOPOLOGY

Source IP addr Dest IP addr LQ ILQ ETX

*** olsr.org - 0.4.10 (Mar 8 2006) ***

(ioctl)Adding route with metric 1 to 192.168.0.24/255.255.255.255 via 192.168.0.24/br0.

(ioctl)Adding route with metric 1 to 192.168.0.178/255.255.255.255 via 192.168.0.24/br0.

--- 00:59:25.53 ----- LINKS

IP address	hyst	LQ	lost	total	NLQ	ETX
192.168.0.178	0.000	0.200	0	2	1.000	5.00
192.168.0.24	0.000	0.300	0	3	1.000	3.33

--- 00:59:25.53 ----- NEIGHBORS

```
IP address  LQ  NLQ  SYM  MPR  MPRS  will
192.168.0.178  0.300  1.000  YES  NO  NO  3
```

--- 00:59:25.53 ----- TOPOLOGY

```
Source IP addr  Dest IP addr  LQ  ILQ  ETX
```

```
*** olsr.org - 0.4.10 (Mar 8 2006) ***
```

```
(ioctl)Deleting route with metric 1 to 192.168.0.178/255.255.255.255 via 192.168.0.24/br0.
```

```
(ioctl)Deleting route with metric 1 to 192.168.0.24/255.255.255.255 via 192.168.0.24/br0.
```

```
(ioctl)Adding route with metric 1 to 192.168.0.178/255.255.255.255 via 192.168.0.178/br0.
```

```
Add route(192.168.0.178): File exists
```

```
(ioctl)Adding route with metric 1 to 192.168.0.24/255.255.255.255 via 192.168.0.178/br0.
```

```
Add route(192.168.0.24): File exists
```

```
.
.
.
.
.
```

Above you can see the SSH screens, with olsrd debug log running and node entries for the neighbouring units.

3.6 Checking OLSRD via a Windows PC

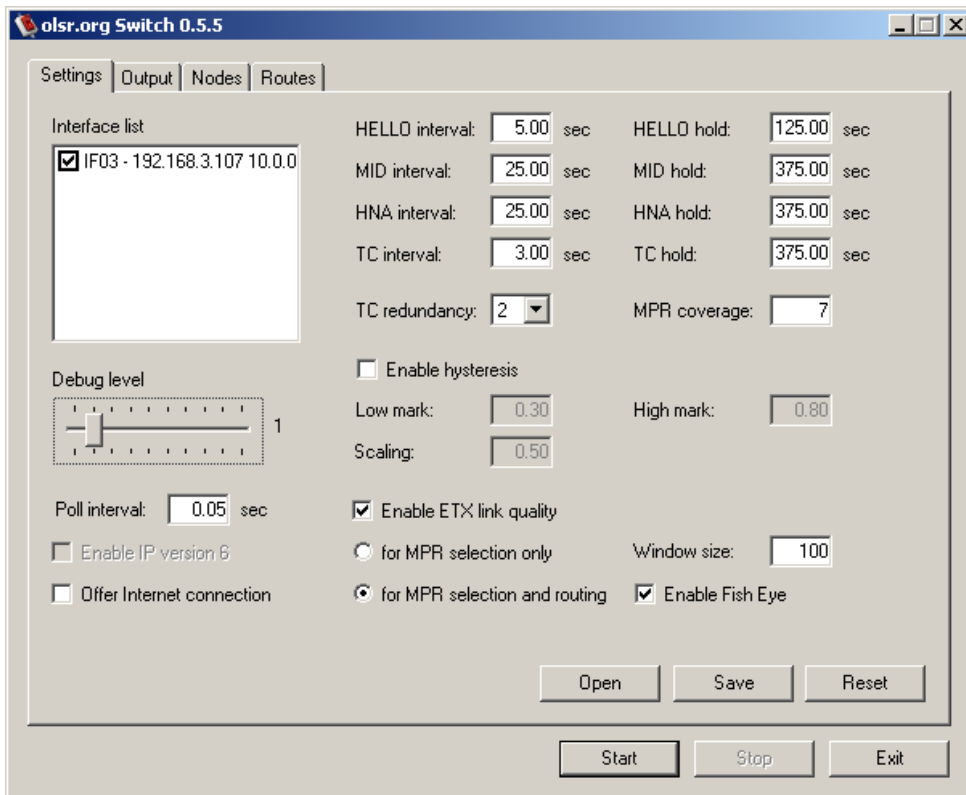
You can further test by running the olsrd switch daemon on any windows PC connected to the mesh network (via LAN port of any meshing unit or wirelessly to any meshing unit).

You can download the olsrd daemon at..

<http://www.olsr.org/releases/0.5/olsrd-0-5-5-setup.exe>

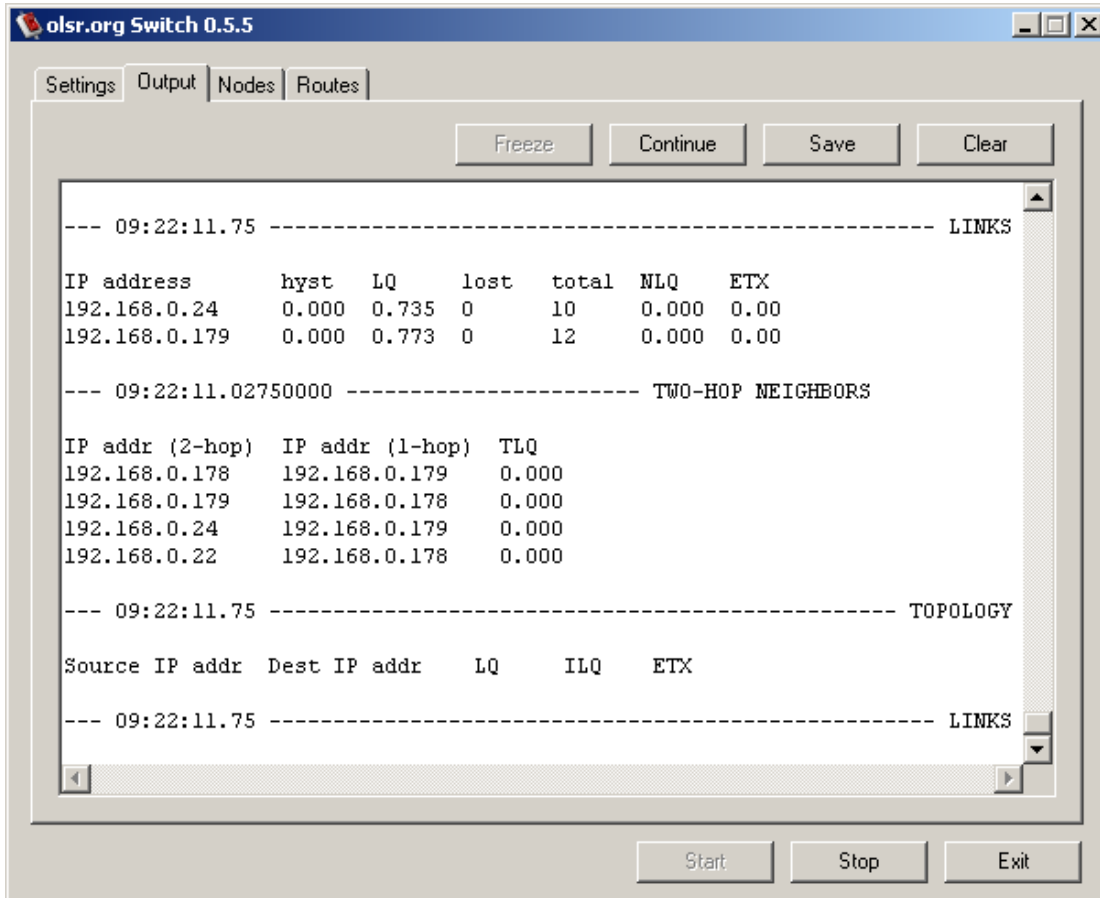
Download and install this application. Next make sure you PC has an IP address in the same subnet as that used for your olsrd nodes.

Run the olsrd switch.



In the Interfaces list you should see the IP address of the lan interface of your PC (I have lot's!). Set the Debug Level to 1 and then click on Start...

Now goto the Output screen:



You can see the other olsr interfaces listed.

In my case

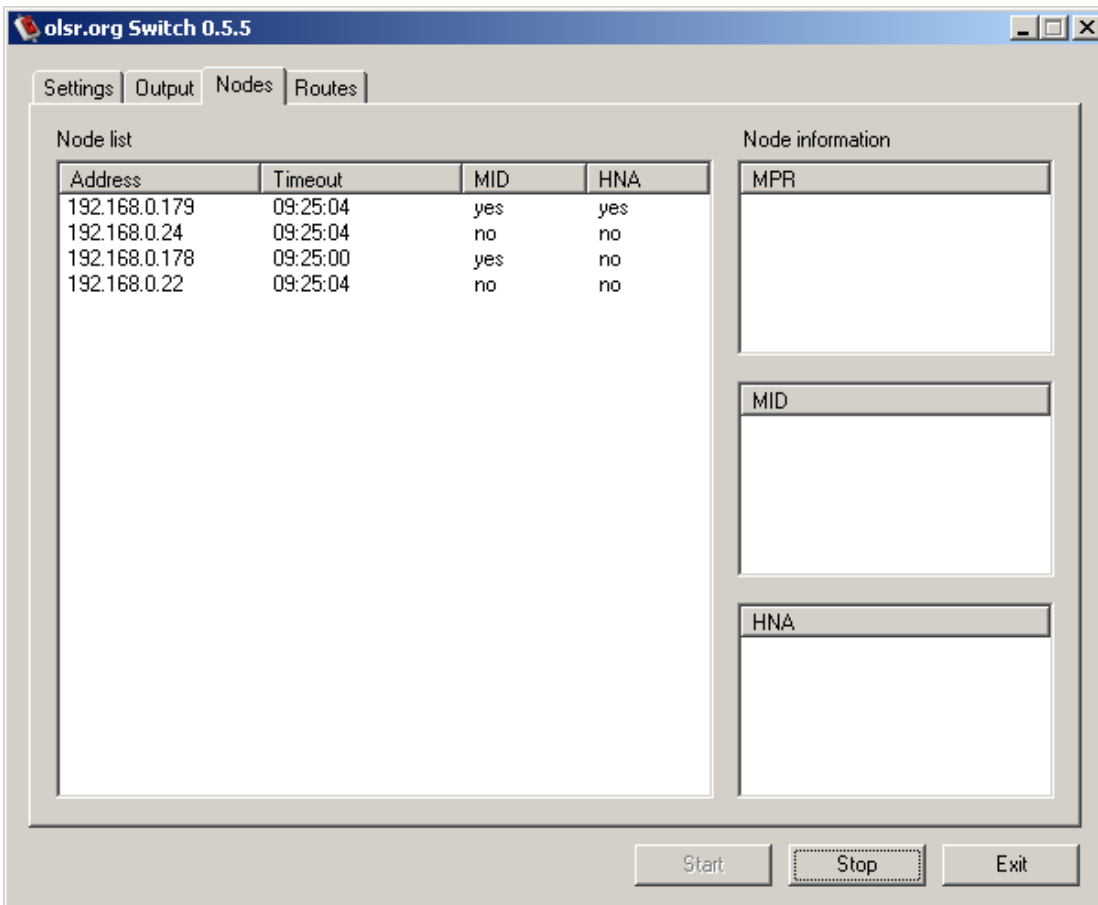
0.22 is LAN of my main node

0.179 is the WLAN of my main node

0.24 is LAN of my main node

0.178 is the WLAN of my main node

Also, if I look at the Nodes page is see...



Notice the extra node addresses. Notice that HNA is showing next the WLAN of my main node. Also notice that the WLAN interfaces of the nodes are now shown as MID. A MID or Multiple interface declaration(MID) is essentially an interfaces on which a node runs OLSR.

If you can't see your nodes then something's wrong 😊

3.6.1 Disabling olsrd via SSH

As mentioned above, when meshing is enabled with debug mode running and you reboot the unit, then the web interface is disabled. So these are instructions for how to use ssh (PuTTY) to disable olsrd. You can then reboot and the web interface will work again.

So, use PuTTY to enter the command screen of the device...

```
192.168.2.1 - PuTTY
login as: root
root@192.168.2.1's password:

BusyBox v1.01 (2006.09.04-14:05+0000) Built-in shell (ash)
Enter 'help' for a list of built-in commands.

# █
```

You can permanently cease olsrd by editing the value set in the flash i.e.

```
#
#
#
#
#
#
#
#
# flash set MESH_ENABLED 0
# █
```

Now save the changes...

```
#
#
#
#
# flash set MESH_ENABLED 0
# save
running set mode
wrote: 21394 bytes
etc Salvo
# █
```

Another option, to temporarily turn olsrd off is to just kill that process using the killall Linux command....

```
#
#
#
#
# killall -9 olsrd
# █
```

Now you can exit PuTTY.

If you have turned olsrd off by changing the flash value then you need to reboot to see the effect (though, remember, if you have turned the setting off then you will need to turn it back on in the WAN page of the web setup). When it comes back up the web interface should be working.

If you just used killall to temporarily stop the process then the web interface should immediately be working.

3.6.2 Olsrd.conf file

```
#
# olsr.org OLSR daemon config file
#
# Lines starting with a # are discarded
#
# This file was shipped with olsrd 0.4.10-cvs
```

```
#

# This file is an example of a typical
# configuration for a mostly static
# network(regarding mobility) using
# the LQ extention

# Debug level(0-9)
# If set to 0 the daemon runs in the background
# Leave as 0 (off) for normal running

DebugLevel      1

# IP version to use (4 or 6)

IpVersion      4

# Clear the screen each time the internal state changes

ClearScreen     yes

# HNA IPv4 routes
# syntax: netaddr netmask
# Example Internet gateway:
# 0.0.0.0 0.0.0.0

Hna4
{
#   Internet gateway: for main node ONLY
   0.0.0.0      0.0.0.0
#   more entries can be added:
#   192.168.0.0 255.255.255.0
}

# HNA IPv6 routes
# syntax: netaddr prefix
# Example Internet gateway:

Hna6
{
#   Internet gateway:
```

```
# ::          0
# more entries can be added:
# fec0:2200:106:: 48
}
```

```
# Should olsrd keep on running even if there are
# no interfaces available? This is a good idea
# for a PCMCIA/USB hotswap environment.
# "yes" OR "no"
```

```
AllowNoInt yes
```

```
# TOS(type of service) value for
# the IP header of control traffic.
# If not set it will default to 16
```

```
#TosValue 16
```

```
# The fixed willingness to use(0-7)
# If not set willingness will be calculated
# dynamically based on battery/power status
# if such information is available
```

```
Willingness 7
```

```
# Allow processes like the GUI front-end
# to connect to the daemon.
```

```
Ipconnect
```

```
{
    # Determines how many simultaneously
    # IPC connections that will be allowed
    # Setting this to 0 disables IPC
```

```
MaxConnections 0
```

```
# By default only 127.0.0.1 is allowed
# to connect. Here allowed hosts can
# be added
```

```
Host          127.0.0.1
#Host         10.0.0.5

# You can also specify entire net-ranges
# that are allowed to connect. Multiple
# entries are allowed

# Net          192.168.2.0 255.255.255.0
}

# Wether to use hysteresis or not
# Hysteresis adds more robustness to the
# link sensing but delays neighbor registration.
# Used by default. 'yes' or 'no'

UseHysteresis    no

# Hysteresis parameters
# Do not alter these unless you know
# what you are doing!
# Set to auto by default. Allowed
# values are floating point values
# in the interval 0,1
# THR_LOW must always be lower than
# THR_HIGH.

#HystScaling     0.50
#HystThrHigh     0.80
#HystThrLow      0.30

# Link quality level
# 0 = do not use link quality
# 1 = use link quality for MPR selection
# 2 = use link quality for MPR selection and routing
# Defaults to 0

LinkQualityLevel 2
```

```
# Link quality window size
# Defaults to 10

LinkQualityWinSize      10

# Polling rate in seconds(float).
# Default value 0.05 sec

Pollrate      0.05

# TC redundancy
# Specifies how much neighbor info should
# be sent in TC messages
# Possible values are:
# 0 - only send MPR selectors
# 1 - send MPR selectors and MPRs
# 2 - send all neighbors
#
# defaults to 0

TcRedundancy      2

#
# MPR coverage
# Specifies how many MPRs a node should
# try select to reach every 2 hop neighbor
#
# Can be set to any integer >0
#
# defaults to 1

MprCoverage 3

# Olsrd plugins to load
# This must be the absolute path to the file
# or the loader will use the following scheme:
# - Try the paths in the LD_LIBRARY_PATH
```

```

# environment variable.
# - The list of libraries cached in /etc/ld.so.cache
# - /lib, followed by /usr/lib

# Example plugin entry with parameters:

#LoadPlugin "olsrd_dyn_gw.so.0.3"
#{
    # Here parameters are set to be sent to the
    # plugin. These are on the form "key" "value".
    # Parameters ofcourse, differs from plugin to plugin.
    # Consult the documentation of your plugin for details.

    # Example: dyn_gw params

    # how often to check for Internet connectivity
    # defaults to 5 secs
# P1Param      "Interval"      "40"

    # if one or more IPv4 addresses are given, do a ping on these in
    # descending order to validate that there is not only an entry in
    # routing table, but also a real internet connection. If any of
    # these addresses could be pinged successfully, the test was
    # succesful, i.e. if the ping on the 1st address was successful,the
    # 2nd won't be pinged
# P1Param      "Ping"          "141.1.1.1"
# P1Param      "Ping"          "194.25.2.129"
#}

# Interfaces and their rules
# Omitted options will be set to the
# default values. Multiple interfaces
# can be specified in the same block
# and multiple blocks can be set.

# !!CHANGE THE INTERFACE LABEL(S) TO MATCH YOUR INTERFACE(S)!!
# (eg. wlan0 or eth1):

Interface "br0" "wlan0"
{

```

```
# IPv4 broadcast address to use. The
# one usefull example would be 255.255.255.255
# If not defined the broadcastaddress
# every card is configured with is used

# Ip4Broadcast          255.255.255.255

# IPv6 address scope to use.
# Must be 'site-local' or 'global'

# Ip6AddrType          site-local

# IPv6 multicast address to use when
# using site-local addresses.
# If not defined, ff05::15 is used

# Ip6MulticastSite     ff05::11

# IPv6 multicast address to use when
# using global addresses
# If not defined, ff0e::1 is used

# Ip6MulticastGlobal   ff0e::1

# Emission intervals.
# If not defined, RFC proposed values will
# be used in most cases.

# Hello interval in seconds(float)
HelloInterval          10.0

# HELLO validity time
HelloValidityTime     100.0

# TC interval in seconds(float)
TcInterval             3.0

# TC validity time
TcValidityTime        30.0
```

```
# MID interval in seconds(float)
MidInterval 5.0

# MID validity time
MidValidityTime 30.0

# HNA interval in seconds(float)
HnaInterval 5.0

# HNA validity time
HnaValidityTime 30.0

# When multiple links exist between hosts
# the weight of interface is used to determine
# the link to use. Normally the weight is
# automatically calculated by olsrd based
# on the characteristics of the interface,
# but here you can specify a fixed value.
# Olsrd will choose links with the lowest value.

# Weight 0

}
```

4 Access Control

The 'van site' described above works very well and is a fairly cost effective means of feeding internet access into each van however one thing to consider is how you are going to charge for the access. One option is to just put it on as a levy on the normal fees (or even give it away free as a site perk) but then you have to worry about locking the wireless network down so that only user that have paid can use the link. A better, more secure option is to use a 'Hotspot router' like the Solwise WAS-102R. The Hotspot router sits between the main, AP in the club house and the current ADSL router (assuming DSL access). It connects via LAN to the outgoing AP and also via LAN to the incoming DSL router and acts as a controller for any internet access going to the AP (and hence to the main site). It is also a wireless router and therefore also gives controlled internet access via local wireless. Using the hotspot you can either manually preset usage accounts or issue 'on demand' ticket or, even configured to do online PayPal charging to the customer. In either case the end user is given a unique, username and password (dynamically created if using on-demand ticket or PayPal) which they must enter from their browser screen when they want to access the internet. Access accounts can be setup to give short term access or longer usage, like the duration of the customers stay or on a weekly basis.