
The Topfield TF5800 and the Asus WL-500g

By Penthes, ChunkyWizard and Jackofall

Version: 1.5, 10 November 2008



Asus WL-500g Premium

Disclaimer

These instructions involve updating the firmware on your Asus WL-500g with a version not supplied by Asus, probably invalidating your warranty. By doing this you are risking irreparably damaging your Asus, and for this no responsibility can be taken. Take particular care over the instructions for flashing the firmware, and in particular the version of the firmware used.

These instructions are provided as-is in the hope that they will be useful to you. There is no guarantee of correctness or completeness.

Comments, corrections, errors and omissions

We would be glad get feedback on this guide, to hear of any problems you have or errors you find, and will do my best to keep the guide up-to-date.

Please post your comments to the toppy.org.uk forums, under the "[Computer connections](#)" section.

Guide History

Version	Date	Comments
0.1	7 January 2006	Initial version
1.0	8 January 2006	Integrated ChunkyWizard's guide. Incorporated comments and corrected errors based on feedback from forums
1.1	9 January 2006	Incorporated Jackofall's guide. Added esmtp for sending emails. Incorporated further feedback.
1.2	12 January 2006	Updated ssh section, included PuTTY example. Added section on how to set up startup script for built-in cron. Amended rt2mei – cron does not need installing, just configuring Changed the rt2mei crontab log to /tmp/var/log/rt2mei to make consistent with other logs Changed email section, there needs to be a blank line before the email body, or the email headers get messed up. Modification to post-boot so that information and error messages from init.d scripts are captured in the post-boot log.
1.3	12 January 2006	Minor typos corrected
1.4	15 January 2006	Automatic BST/GMT changeover (important for cron) Corrected the "- pastes as ." problem when pasting text from the guide Instructions for how to use crontab with nano Added information on vi and nano text editors, and links to instructions Setting the default crontab editor to be nano Other corrections and clarifications
1.5	9 November 2008	Updated for Oleg's firmware version 1.9.2.7-10 DNS is now configurable in the web interface, removed instructions for scripts to set the DNS server address Web interface screen captures updated Other corrections, clarifications and hints and tips supplied by users of the previous version of the guide

Document conventions

Entering commands

You will be typing commands for the Asus in a telnet session, using MS Windows telnet, PuTTY or the telnet client of your choice (see Accessing your Asus using Telnet on p7 for more on this).

Commands that you type are shown as follows:

```
mkdir /opt/etc/samba
ls '/a path to a long directory name to show the format of text wrapping in
    this document/*'
touch /opt/etc/samba/smb.conf
```

In your telnet session, you will see a prompt something like [admin@WL500gP root]\$ -- type the commands at this prompt, terminating each command with the <enter> key.

If a new line of text has been started because of the formatting of this document, the new line will be shown indented. This does *not* mean that you press <enter> at this point, nor should you be doing any indenting when you type the command in.

Editing files

You will also be creating and editing files on the Asus. To do this you need to use an editor (such as vi or nano, see Text editors: vi and nano on p8), again through a telnet session.

Files that you are editing are shown in this document as follows, with the heading showing the full path and name of the file.

/opt/etc/init.d/S00set-dns.sh

```
#!/bin/sh
#
# S00-set-dns.sh
#
# add the dns server to resolv.conf

echo '# resolv.conf' > /etc/resolv.conf
echo '# just a long comment to show how long lines of text wrap in this
    document' >> resolv.conf
echo "# generated by $0" >> /etc/resolv.conf
echo 'nameserver 10.1.1.1' >> /etc/resolv.conf
```

Where lines of text have been wrapped due to the formatting of this document, they are shown indented (so in the example above, the line starting echo '# just a long comment, and the following indented line, are in fact all part of the same line). Do not start a new line in your editor at this point, and don't add any indenting of your own.

Files used in this document

All of the files that you will be creating in this document are available in a zip file from <http://www.tippy.org.uk/~chunkywizard/WL500g/TippyAndAsusWL-500g.zip>.

The folder structure in the zip file matches the directory structure used on the Asus.

Take care if you use these files as a starting point, make sure you follow the guide carefully, as some of these files will need editing to suit your own configuration, and therefore won't work if you use them just as they are.

Contents

Why get an Asus WL-500g?	5
What is involved?	5
The PC, Windows and your home network	5
The Hardware.....	5
Asus firmware version.....	6
Accessing your Asus	7
Upgrading the firmware	8
Basic configuration using the web interface.....	10
Adding a USB key: partition, format, mount on boot	15
Using the built-in FTP server and Samba server	19
FTP access to files on the Toppo using ftpd-topfield.....	20
Windows file sharing access to the Asus using Samba	23
Command-line access to the Toppo using Puppy	24
Set up cron for scheduling tasks	24
Automated download of Radio Times EPG using rt2mei	26
Secure access using ssh: external access to your Asus	29
Sending emails from your Asus	31
Other hints and tips.....	33
FAQ	35
Thanks	37
References	37

Why get an Asus WL-500g?

The Toppo doesn't have an Ethernet port or any wireless capability, out-of-the-box the only way to connect it to a PC is through its USB port.

If you wish to add your Toppo to your wireless network, you will need a wireless device that supports attached USB storage, such as the Asus WL500. There are various flavours of these but the most currently available is the WL-500g Premium, or WL500gP for short, and that is the model that is used in this guide, though most of the instructions also apply to the WL-500g Deluxe.

The Asus comes with two external USB ports, one of which you can use for connecting to your Toppo. So rather than having to connect a USB lead from your Toppo to your PC, with the Asus it's possible to connect wirelessly to the Toppo, avoiding trailing cables all around the house.

The Asus is a small Linux-based (BusyBox) device, and with the addition of Oleg's firmware you can install additional software on it.

You will also need a USB key, which plugs into the second USB port. This provides the extra storage necessary for installing the various software packages; we would recommend at least 128MB.

By following the instructions in this guide, it will be possible for you to transfer recordings from the Toppo to your PC for archiving and burning to DVD, to access files on the Asus itself using Windows File and Printer Sharing (SMB), and to have a daily automated download of the Radio Times EPG to the Toppo for use with the MyStuff TAP.

What is involved?

To get the Asus to perform these functions you need to add functionality to it. This involves upgrading the firmware and loading other packages on. This is technical, but there should be enough information out there (and in this guide) so you are following line-by-line instructions to complete this task. The tricky bit is when it does not work, and then you may need to ask on toppy.org.uk forums for help.

How long does it take?

It took the authors of this guide about 3 days on and off (mostly on) to get it working. Should everything run smoothly it should take a lot less, as we didn't have this guide to work from!

The PC, Windows and your home network

This guide was written for a PC running Windows 2000 or Windows XP. It is assumed you are on an ADSL broadband connection with a wireless ADSL modem/router, though it is also possible to use the Asus with a wired Ethernet connection. And if your ADSL modem/router is not wireless, it's also possible to add a second wireless router to provide the link to the Asus, though you will have to do some messing around with routing tables.

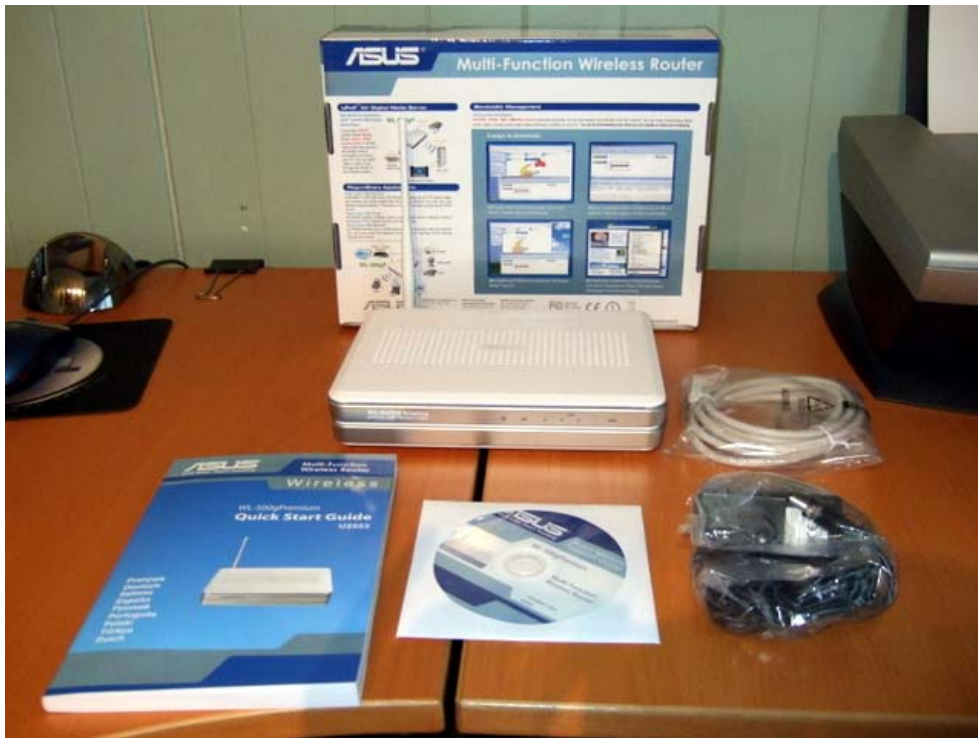
It's based on a home network with a network address of 192.168.1.0 and a subnet mask of 255.255.255.0 (192.168.1.0/24). The Asus has a default IP address of 192.168.1.1. If your network uses different network addresses you will need to modify the appropriate values in this guide.

The Hardware

The hardware you will need is

- a) An Asus WL-500g (the Premium was used for this guide, most of the instructions are also applicable to the Deluxe)
- b) A USB key, minimum size 128MB. A 1GB key was used for this guide, but all of the software installed in the making of this guide will fit into 64MB of storage.

So what do you get for your money? See below.



As you can see, the Asus is about A5 size. It comes with a Quick Start guide (which you will not need), a CD (containing a manual and some tools which hopefully you will not need), and an Ethernet cable and power supply which you will need. If you got your router from Komplett then they ship US versions and a plug adapter. This works fine but might mean you have to rearrange your plugs for it to fit, see below.



Asus firmware version

The instructions in this guide have been tested with the Topfield TF5800, the Asus WL-500gP and Oleg's firmware version 1.9.2.7-10. The Firmware upgrade, basic configuration, USB key and ftpd-topfield sections were all taken from Jackofall's original "[Connecting an Asus WL500g deluxe to your network](#)" guide. Great care must be taken in selecting the correct firmware version for your model.

Accessing your Asus

Your initial access to the Asus will be through a wired Ethernet connection, until you have the wireless connection set up. Plug the Ethernet cable into the LAN1 connection on the Asus.

You will need to make sure that your PC and the Asus are on the same IP address range (subnet), by default this is the address range from 192.168.1.1 to 192.168.1.254. The Asus itself has a default IP address of 192.168.1.1, and this is the one that is used in this guide. Make sure your existing ADSL modem/router's IP address, and any other PCs on your network, don't conflict with the address you use for the Asus.

If you are connecting your PC straight to the Asus with an Ethernet cable to do the initial configuration, you will need to assign a static IP address to your PC, eg 192.168.1.2.

If you need help in setting up a static IP on your PC visit <http://www.portforward.com/networking/static-xp.htm>

Accessing your Asus through a web browser

The normal way of accessing your Asus is through a web browser, with the IP address of your Asus as the address (the default is 192.168.1.1).

This gives you the web configuration interface of the Asus, and you'll be using that to set up the basic configuration.

Accessing your Asus using Telnet

In most of the procedures in this guide you will need to enter Linux commands and edit files.

This is done by using telnet – this gives you command-line access to the router. There are a number of telnet clients you can run on your PC to do this, and Windows comes with a command-prompt version of Telnet built in. PuTTY comes highly-recommended as a telnet client.

Please note that if you want to access your Asus using telnet before you have installed Oleg's firmware, you will need to turn it on in the web interface. Enable it under System Setup -> Services.

If you get as far as installing the dropbear sshd server, you can also access the router using an ssh client (PuTTY is an ssh client as well as a telnet client).

MS Windows Telnet

To start Telnet, open a command prompt window (you can do this by clicking on Run from the Start button, and entering `cmd`, or you can find it elsewhere in the start menu).

Then type

```
telnet 192.168.1.1
```

and enter your user id and password (the defaults are admin and admin).

Hints and tips:

- You can paste stuff into the telnet window by positioning the mouse pointer over it and right-clicking. Note that if you are using the vi editor you will need to press `i` to get into insert mode first. If you paste multiple lines of text that contain indentations into vi, vi does some strange things with the indentations, so you are probably better off pasting line-by-line.
- You can copy text from the telnet window by selecting it by clicking and dragging with the mouse, and then pressing the Enter key. The selected text will be highlighted in white (unless you have changed the default command prompt colours).
- If you get into a mess, and nothing seems to be responding, type `ctrl-]` and type quit at the Microsoft Telnet> prompt. This will close the telnet session and return you to the Windows command prompt

PuTTY

PuTTY is an excellent Windows telnet client. You can get it from <http://www.chiark.greenend.org.uk/~sgtatham/putty/>

Hints and tips:

- a) Connect by entering the Asus IP address (default 192.168.1.1) in the Host Name box. Select the telnet protocol. Enter your user name and password (default admin and admin) when prompted.
- b) You can paste into the PuTTY window using right-click.
- c) You can copy text from the PuTTY window simply by selecting it (mouse click-and-drag). The text is automatically copied to the clipboard.

Text editors: vi and nano

Whether you choose to use MS Windows Telnet, PuTTY or some other Telnet or SSH client, a lot of the instructions in this guide ask you to edit files.

This involves using a text editor once you are in a telnet session.

Oleg's firmware comes with a built-in text editor, called vi. It's pretty basic in terms of user interface, but it does work. Alternatively you can install nano, which is a bit more user-friendly.

You'll need to have got past the section Macsat: Installing basic software (p17) before you can do this.

To install nano, in a telnet window type

```
ipkg install nano
```

For information on editing files using vi and nano, try

vi: <http://unixhelp.ed.ac.uk/vi/index.html> and <http://www.cs.rit.edu/~cslab/vi.html#A1.1>

nano: <http://www.nano-editor.org/dist/v2.0/nano.html>

Important: Note that nano will, by default, add in extra line breaks if you paste in lines over 40 characters. So if you are copying and pasting in from this guide, it's best to specify the `--nowrap` option, eg for editing the post-boot script use the following command:

```
nano --nowrap /usr/local/sbin/post-boot
```

Upgrading the firmware

Upgrading the firmware consists of replacing the Asus-provided firmware with Oleg's firmware. This allows you to plug a USB key into the machine, mount it and install additional software on it.

It is recommended that you avoid making any configuration changes (eg user name, password, network configuration) until you have installed Oleg's firmware.

You are also recommended to make sure you have a copy of the official Asus firmware, and the firmware restoration tool. Both of these can be obtained from <http://uk.asus.com/>, and can hopefully be used to restore the Asus to its original state, either using the web interface or by using Recovery Mode.

Downloading the firmware

The latest version of Oleg's firmware is available from <http://oleg.wl500g.info/#latest>. The version used in this guide is 1.9.2.7-10 and we recommend you use this version. People have ended up destroying their Asus using some of the earlier versions, so if you plan to use a version other than this, do your own research to make sure it will work with your model of the Asus WL-500g first. And make sure you download the correct version for your model – as there are different versions for different models.

Note that 1.9.2.7-9 (not -10) is the last version supporting some versions of the WL500-g, so check carefully at Oleg's firmware page that you are using a firmware version suitable for your model. The instructions in this guide should work for -9 and -10 versions of the firmware..

The firmware is in a zip file, you'll need to unzip it and extract the .trx file.

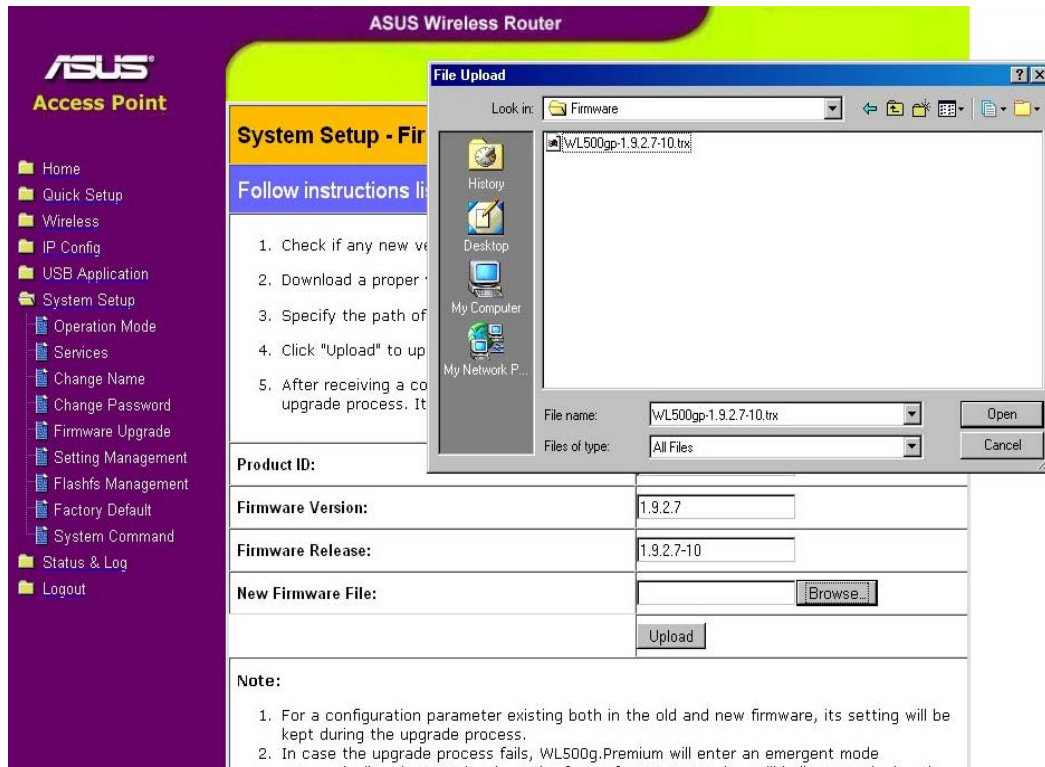
Installing the firmware using the web interface

You can use the web interface of the Asus to upload the firmware.

If you have any problems with this, there is an alternative method that involves putting the Asus into Recovery Mode, which is covered in the next section.

It is recommended that you do this with a wired Ethernet connection.

Browse to <http://192.168.1.1/> (or your chosen Asus IP), enter your login details and navigate to System Setup -> Firmware Upgrade



Click on browse and find the previously downloaded .trx file, click open then upload. The Asus will then show a progress bar and reboot when upgrade is complete.

Installing the firmware using Recovery Mode

It is recommended that you do this with a wired Ethernet connection.

To install the firmware:

- Unplug the power cable from the Asus, and leave it for at least 20 seconds
- Press and hold the "Restore" button on the router
- Plug the power cable back into the Asus.
- Continue to hold the button in until the power light starts blinking
- In a command prompt window on your PC navigate to the directory containing your firmware and type

```
tftp -i 192.168.1.1 put WL500gp-1.9.2.7-7f.trx
```

Note that if you have changed the IP address of your Asus, you should replace 192.168.1.1 with the new IP address. You should replace WL500gp-1.9.2.7-7f.trx with the filename of the firmware you downloaded, if it is different.

The Asus should respond with a message something like:

```
Transfer successful: 7938048 bytes in 28 seconds, 283501 bytes/s
```

The router will then reboot, this can take a few minutes, wait until the power light is steady.

Upgrading firmware

If you are upgrading your firmware from a previous version, it is recommended you return your Asus to the factory settings, either before or after uploading the new version. This can be done from the web interface, under System Setup -> Factory Default

Basic configuration using the web interface

The web interface is used to perform the basic LAN and wireless setup

This covers configuring the Asus in Hybrid mode, so it should be able to connect to your main wireless router using WDS, and also allow wireless connections to itself. We also cover using the Asus in Station mode, which means it connects to your main ADSL modem/router in the same way your PC does, in case your main ADSL modem/router does not support WDS, or you have problems connecting through that route.

Using your browser navigate to the Asus current IP of <http://192.168.1.1/>. You will need to enter the default username and password of admin and admin (you can change these once you have completed setting up the Asus)

Operation mode

First we are going to change the operation mode of the Asus, changing it to an access point so using the menu on the left of the screen navigate to System Setup -> Operation Mode

ASUS Wireless Router

ASUS
Access Point

Home
Quick Setup
Wireless
IP Config
USB Application
System Setup
Operation Mode
Services
Change Name
Change Password
Firmware Upgrade
Setting Management
Flashfs Management
Factory Default
System Command
Status & Log
Logout

System Setup - Operation Mode

WL500g.Premium supports three operation modes to meet different requirements from different group of people. Please select the mode that match your situation.

<input type="radio"/> Home Gateway	In this mode, we suppose you use WL500g.Premium to connect to Internet through ADSL or Cable Modem. And, there are many people in your environment share the same IP to ISP. Explaining with technical terms, gateway mode is , NAT is enabled, WAN connection is allowed by using PPPoE, or DHCP client, or static IP. In addition, some features which are useful for home user, such as UPnP and DDNS, are supported.
<input type="radio"/> Router	In Router mode, we suppose you use WL500g.Premium to connect to LAN in your company. So, you can set up routing protocol to meet your requirement in office. Explaining with technical terms, router mode is, NAT is disabled, static routing protocol are allowed to set.
<input checked="" type="radio"/> Access Point	In Access Point mode, all 5 Ethernet ports and wireless devices are set to locate in the same local area network. Those WAN related functions are not supported here. Explaining with technical terms, access point mode is, NAT is disabled, one wan port and four lan ports of WL500g.Premium are bridged together.

Apply

Click on Access Point and click on apply.

Check the ASUS is set to the correct time zone in

IP Config -> Miscellaneous

IP Config - LAN

We are now going to change the IP address of the Asus, so navigate to
IP Config -> LAN

ASUS Wireless Router

ASUS Access Point

Home
Quick Setup
Wireless
IP Config
LAN
IPv6
SNMP
Miscellaneous
USB Application
System Setup
Operation Mode
Services
Change Name
Change Password
Firmware Upgrade
Setting Management
Flashfs Management
Factory Default
System Command
Status & Log
Logout

IP Config - LAN

LAN IP Setting

Host Name:	MyAsus
Get IP Automatically?	<input type="radio"/> Yes <input checked="" type="radio"/> No
IP Address:	192.168.1.151
Subnet Mask:	255.255.255.0
Default Gateway:	192.168.1.1
DNS Server:	192.168.1.1

Restore Finish Apply

Restore:	Clear the above settings and restore the settings in effect.
Finish:	Confirm all settings and restart WL500g.Premium now.
Apply:	Confirm above settings and continue.

Here we need to make the following changes

Get IP Automatically?	No
IP Address	see note 1
Subnet Mask	255.255.255.0
Default Gateway	see note 2
DNS Server	see note 3

- 1) The IP address you choose for the Asus will depend on what IP addresses you are using for other devices on your network, particularly your main ADSL modem/router. If your main ADSL modem/router is on a different IP address from 192.168.1.1, you can leave the Asus on the default IP address. Otherwise you need to choose a different one, eg 192.168.1.151 – if you change the IP address here, make sure that this is the IP address you use for the rest of this guide.
- 2) This should be the IP address of your main ADSL modem/router.
- 3) You can use the IP address of your main ADSL modem/router as the DNS server, as most routers are configured to obtain ISP DNS addresses automatically and forward DNS queries to them. However you may experience timeout issues with ftpd-topfield, in which case it's probably better to directly enter the IP address of your ISP DNS servers here.

Next click Apply and then Finish you will then be taken to the following screen.

Save & Restart

Save&Restart will save all setting you have changed to WL500g.Premium and restart it. Please click **SaveRestart** button to continue.

Save&Restart

Click Save&Restart and then the ASUS will then reboot.

Wireless setup

If you changed the IP address in the previous section, you now need to point your browser to the new IP address.

Enter your login details as before, and navigate to

Wireless -> Interface

ASUS Wireless Router

Access Point

Wireless - Interface

SSID:	YOUR_SSID
Channel:	3
Wireless Mode:	Auto <input type="checkbox"/> 54g Protection
Authentication Method:	WPA-Auto-Personal
WPA Encryption:	TKIP+AES
WPA Pre-Shared Key:	*****
WEP Encryption:	None
Passphrase:	
WEP Key 1 (10 or 26 hex digits):	*****
WEP Key 2 (10 or 26 hex digits):	*****
WEP Key 3 (10 or 26 hex digits):	*****
WEP Key 4 (10 or 26 hex digits):	*****
Key Index:	2
Network Key Rotation Interval:	0

Restore Finish Apply

Here you will need to enter the following information

SSID:	The same as your existing AP
Channel:	The same as your existing AP
Authentication Method:	The same as your existing AP
WPA Encryption, WPA Pre-Shared Key:	The same as your existing AP
or	
WEP Encryption:	The same as your existing AP
Passphrase or WEP keys:	The same as your existing AP

Click Apply and then navigate to
Wireless -> Bridge

ASUS Wireless Router

ASUS
Access Point

Home
Quick Setup
Wireless
Interface
Bridge
Access Control
RADIUS Setting
Advanced
IP Config
USB Application
System Setup
Status & Log
Logout

Wireless - Bridge

Wireless bridge (also known as Wireless Distribution System or WDS) function allows you to connect to one or many APs through wireless.

AP Mode: Hybrid

Channel: 3

Connect to APs in Remote Bridge List? ☒ Yes ☐ No

Allow anonymous? ☐ Yes ☒ No

Remote Bridge List

MAC Address
001234123412

Restore Finish Apply

Here you will need to enter the following information

AP Mode: Hybrid (1)
Channel: As Before
Connect to AP's in remote bridge list? Yes
Allow anonymous? No

1) If you are going to use Station mode, instead of WDS, then choose WDS Only here

You will need to add the MAC Address of the AP you wish the Asus to become a client of enter it in the format 000F66251CAE omitting any : and click Add.

Click Apply

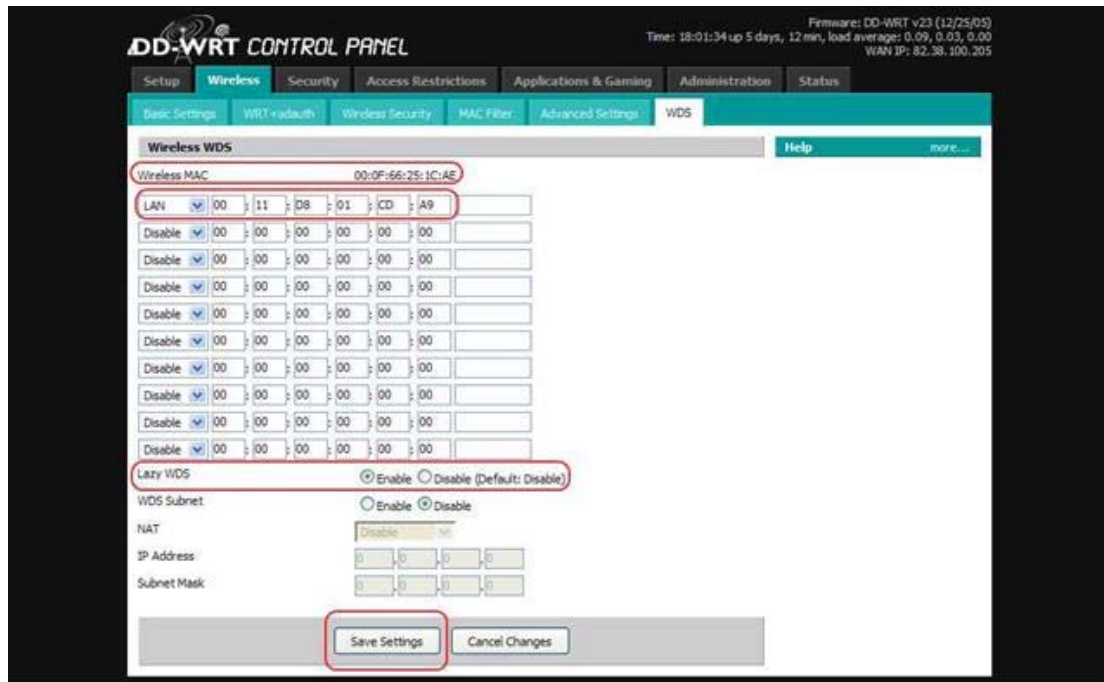
If you are going to use Station mode, instead of WDS, you will also need to navigate to Wireless -> Advanced, and set the Extended Mode to station, and again click Apply.

Click Finish as before you will be taken to the Save&Restart page.

Click Save&Restart and then the ASUS will then reboot

(Optionally you can apply any MAC address filtering on the Wireless -> Access Control page)

You will now need to go to the web interface of your Access Point. in this case we are using a Linksys WRT54G running the DD-WRT firmware. Please refer to the documentation relating to your individual Access point regarding WDS.



Here you will need to enter the MAC address of the Asus (you can find this on the white sticker on the side of the product), change the setting to LAN and in this case we also enabled Lazy WDS and clicked Save Settings. The Asus should now show as a WDS Node in your other AP's status page.

Now disconnect your CAT5 cable, and if you previously assigned a static IP to your PC change the connection settings to their previous value.

Using a wireless client go to the IP Address of the Asus and ensure that you can connect.

You can now change the username and password settings in the System Setup and associated sub menus.

Once you are happy that all is working as it should be Navigate to
System Setup -> Settings Management

And follow the instructions for saving a config file that you can use at a later date if needed.

Adding a USB key: partition, format, mount on boot

This is covered by the macsat guide at <http://www.macsat.com/macsat/content/view/22/29/>.

For the most part we will be following the macsat guide, but there are some minor differences, and some additional information to help you through.

Here we are into the techy stuff. You might want to refer back to Accessing your Asus using Telnet (p7) and Text editors: vi and nano (p8) for information on how to connect and issue the commands in this and subsequent sections, and how to edit files on your Asus.

Macsat steps 1, 2, 3 and 4: Adding the key and preparing for partitioning

You need to follow these instructions carefully. To ensure everything goes smoothly, it's best to turn the router on without the USB key inserted, then plug it in once the Asus has booted up. Then follow the macsat instructions.

Macsat step 1 to 4: Logging in to the router and unmounting the USB key

Follow these instructions as per the macsat guide.

If you have problems at step 4 – ie the umount command gives an error message, then as well as under USB Application, FTP Server (or change the port number)

```
killall stupid-ftp
```

try

```
killall smbd  
killall nmbd
```

in case the built-in samba server is also started. Then try and umount again with

```
umount /tmp/harddisk/
```

If you still get an error message, then try the following:

- d) Reboot your router *leaving the USB key plugged in*

```
reboot
```

- b) See if the USB key is visible by typing

```
ls /dev/discs
```

If this shows `disc0`

then skip step c and proceed with the next section. You do not need to umount the USB key.

However, if you get an error message

```
ls: /dev/discs: No such file or directory
```

then carry out step c

(If FTP or Samba are enabled in the web interface, the modules inserted in c) below are loaded automatically and you won't get an error.)

- c) Insert the modules necessary to access the USB key, and check it is now visible by typing:

```
insmod scsi_mod && insmod sd_mod && insmod usb-storage  
ls /dev/discs
```

You should now see `/dev/discs/disc0` listed

As the normal Asus automount has not happened, there is no need to unmount the key.

Macsat step 5 to 8: partitioning, formatting, rebooting and mounting

You can now proceed with partitioning and formatting, steps 5 and 6 in the macsat guide.

The partitions you set up will depend on the size of your USB key. These instructions are for a 1GB USB key, the macsat guide is for a 256MB key.

You can determine what partition sizes to use by typing

```
fdisk /dev/discs/disc0/disc -l
```

The first three lines will show the geometry and size, mine shows

```
Disk /dev/discs/disc0/disc: 1009 MB, 1009254400 bytes
64 heads, 32 sectors/track, 962 cylinders
Units = cylinders of 2048 * 512 = 1048576 bytes
```

So the total size is shown as 1009MB, deducting 69MB for the swap partition leaves 940MB for the remainder. These calculations are not 100% accurate. In fact these figures result in a swap partition size of 65MB, so when you are doing your own calculations you might want to add on 5-10MB to the swap partition size you want just to make sure it ends up big enough.

This results in a partition setup of:

part1: 940MB e2fs partition (the main Asus partition)

part2: remainder swap partition, currently not used, I might remove this at some point

The fdisk commands for creating these partitions are much the same as the macsat guide, but with a different size for the first partition (don't type the comments in brackets!).

```
d (delete existing partition, if using a key that has been used before,
    enter each partition number in turn, p to check t)
n (new partition)
p (primary partition)
1 (partition number)
<enter> (first cylinder, 1 is default)
+940M (size)
n (new partition)
p (primary)
2 (number)
<enter> (accept default start cylinder)
<enter> (accept default end cylinder)
( check with p here to display what you have set up)
t (partition type)
2 (partition 2)
82 (linux swap)
w (write changes)
```

Now format the partitions as per macsat step 6 and reboot the router as per 7.

When you get to step 8, first check that the USB key is visible, and if not insert the necessary modules:

a) See if the USB key is visible by typing

```
ls /dev/discs
```

If this shows

```
disc0
```

then skip step c

However, if you get an error message

```
ls: /dev/discs: No such file or directory
```

then carry out step c

- c) Insert the modules necessary to access the USB key, and check it is now visible by typing:

```
insmod scsi_mod && insmod sd_mod && insmod usb-storage  
ls /dev/discs
```

You should now see /dev/discs/disc0 listed

Now proceed to mount the disk as per step 8, in my case as I am not (yet) using the swap partition, the command

```
swapon /dev/discs/disc0/part2
```

was omitted.

Macsat: Installing basic software

Install ipkg as per macsat's "Installing Basic Software", then proceed with the rest of this section in the macsat guide.

Macsat: Configuring the router to mount on boot

You can now set up the post-boot script as per macsat's "Configuring the router to mount on boot", but use the post-boot script in this guide, which differs in the following respects:

- 1) As the modules scsi_mod, sd_mod and usb-storage (necessary for accessing the USB key) are not inserted by default if FTP and Samba are disabled, there is a section at the beginning to check and load them if necessary.
- 2) The command to mount the swap partition, this line was commented out (currently not felt to be necessary, but as the partition is there it can be included later if it turns out to be needed).
- 3) Logging has been added, to /tmp/var/log/post-boot, so if there are problems with the boot process you can check this file. This will also capture any error messages generated by the /opt/etc/init.d scripts.

The post-boot script follows.

Make sure that you

```
flashfs save  
flashfs commit  
flashfs enable
```

after creating this file, or you will loose your changes when you reboot!

/usr/local/sbin/post-boot

```
#!/bin/sh
#
# post-boot
# 1. insert modules for USB storage, if needed
# 2. mount part1 of usb drive as /opt
#

log=/tmp/var/log/post-boot
echo `date '+%F %T'` $0 started >>$log

# test if USB disc has been attached
# if not then insert needed modules
if [ ! -d /dev/discs ]
then
    echo `date '+%F %T'` /dev/discs not found, inserting usb modules >>$log
    insmod scsi_mod 2>>$log
    insmod sd_mod 2>>$log
    insmod usb-storage 2>>$log
    sleep 5s
fi

# check /dev/discs now visible
if [ ! -d /dev/discs ]
then
    echo `date '+%F %T'` error: /dev/discs not found, check USB key inserted
    >>$log
else
    echo `date '+%F %T'` /dev/discs found, USB key present >>$log
fi

# Wait for /opt to mount
mount /dev/discs/disc0/part1 /opt 2>>$log
i=0
while [ $i -le 30 ]
do
    if [ -d /opt/etc ]
    then
        echo `date '+%F %T'` USB key partition mounted as /opt >>$log
        break
    fi
    sleep 1
    i=`expr $i + 1`
done

# check it worked
if [ ! -d /opt/etc ]
then
    echo `date '+%F %T'` error: could not mount USB key partition >>$log
fi

# Activate swap
# removed, not currently using swap partition
# swapon /dev/discs/disc0/part2

# Run all active services (all in /opt/etc/init.d starting with S)
echo `date '+%F %T'` starting rc.unslung >>$log
/opt/etc/init.d/rc.unslung >>$log 2>&1

echo `date '+%F %T'` $0 finished >>$log
```

Take care with lines that have split onto two because of the formatting in this document – these should be fairly obvious as the second half of the line is indented significantly. You can

copy/paste straight from this guide. Also note the backwards-apostrophe character used in some of the "echo" statements. This is probably on the top left key on your keyboard.

After you have rebooted as per the macsat guide, you can check for any problems with

```
cat /tmp/var/log/post-boot
```

Using the built-in FTP server and Samba server

Oleg's firmware contains both an FTP server (stupid-ftpd) and Samba, which can be used to access files on the USB key only (they can't be used to access the Toppo, nor can they be used to access files on the Asus itself – for this you should refer to the instructions for installing ftpd-topfield and Samba).

These are configured from the web interface.

Once you have set up a post-boot script these services only start if the USB key is plugged in *after* the Asus has booted up, which means any additional software on the USB key won't run.

And if you then reboot the Asus with the key left in, the built-in FTP and Samba services will not start this time around.

(If you haven't set up the post-boot script, or haven't made it executable, then the built-in services will start.)

This means you can leave both the built-in FTP server and Samba server enabled, as there should be no conflict with ftpd-topfield and the custom Samba configuration.

Warning: some people have had problems transferring large files with ftpd-topfield when the built-in FTP server is turned on.

In normal operation (booting up with the USB key present) the built-in services won't start, and you can instead use ftpd-topfield to browse the Toppo's files, and a custom Samba configuration to browse the Asus filing system, if you have set one up.

It is also safe for both ftpd-topfield and the built-in ftp server to use the same port (21) as they will never be running at the same time.

Using the built-in FTP server

To enable the FTP server in the web interface:

- 1) Open a browser and put in your Asus IP address, and enter your user name and password when prompted.
- 2) Click on USB Application on the left hand side
- 3) Click on FTP Server on the left hand side
- 4) Set the Enable FTP Server option to Yes
- 5) You also have options to allow anonymous users, and to allow the Super User (ie the admin user), and the ability to add other user accounts. I just allow anonymous access.
- 6) Use the Apply button to apply your changes, then click on the Finish button, and when the next page appears click on the button to restart the Asus

You can now use your favourite FTP client to browse the directories and files on the USB

Remember: the FTP server will only start if you boot up the Asus *without* the USB key present, and then insert it.

The FTP root directory corresponds to the root of the USB key. Therefore the folder you normally see as /opt/etc if you boot with the USB key present will now be visible as /etc.

Using the built-in Samba server

To enable the Samba server in the web interface:

- 1) Open a browser and put in your Asus IP address, and enter your user name and password when prompted.
- 2) Click on USB Application on the left hand side

- 3) Click on Samba on the left hand side
- 4) Set the option Samba File Shares to Manual Share List
- 5) Set the option Default Access Level to Read/Write
- 6) Check the DOS Codepage parameter, you probably want this set to 850 (Western Europe), so change it to this if necessary
- 7) Change the Workgroup name to the workgroup you use on your Windows PC
- 8) Under manual Network Shares list, add a share with Directory /opt, Share Name opt and Access Level Read/Write and use the Add button to add it to the list
- 9) Use the Apply button to apply your changes, then click on the Finish button, and when the next page appears click on the button to restart the Asus.

Remember: the Samba server will only start if you boot up the Asus *without* the USB key present, and then insert it.

The shared folder(s) on the Asus are configured in the Manual Network Shares List, so you will see a share called opt corresponding to the folder /opt. You can add additional shares as necessary.

FTP access to files on the Toppo using ftpd-topfield

ftpd-topfield allows you to see the files on your Toppo through an ftp client on your PC.

ftpd-topfield is less well known than the puppy application. However, although it is slightly slower than puppy it does allow a GUI interface on the Windows PC by using any suitable ftp program. In comparison puppy is only command line, and only gives you access from the Asus itself.

ftpd-topfield supports only a single active connection at a time, so if your ftp client supports multiple connections make sure you disable this.

Installation

Telnet into the ASUS as before and log in with your username and password.

Install the ftpd-topfield application by typing:

```
ipkg update
ipkg install ftpd-topfield
```

The first command gets the latest list of packages available and the second installs the latest version of ftpd-topfield. Note the same pair of command will also update any previous installed version too.

When a new version of ftpd-topfield is available you can upgrade with the ipkg command the same as for a fresh install. Note this will stop the current ftpd-topfield, install the new one, and restart ftpd-topfield so there is nothing else you need to do. If the install objects then you can also remove ftpd-topfield by typing

```
ipkg remove ftpd-topfield
```

This will start you with a clean slate for the install

Configuration

The command

```
ipkg files ftpd-topfield
```

Will return two pieces of information, the program location and the startup script. The results will be something like this:

```
Package ftpd-topfield (0.6.4-3) is installed on root and has the following
files:
/opt/sbin/ftpd-topfield
/opt/etc/init.d/S67ftpd-topfield
```

To find out the program options enter the command:

```
/opt/sbin/ftpd-topfield --help
```

This will return something like this:

```
Usage: /opt/sbin/ftpd-topfield [OPTION] ...
Internet File Transfer Protocol server.

  -A, --anonymous-only      Server configure for anonymous service only
  -b, --turbo                Default to turbo mode on for transfers
  -D, --daemon               Start the ftpd standalone
  -d, --debug                Debug mode
  -E, --eplf                 Send listing in EPLF mode
  -l, --logging              Increase verbosity of syslog messages
  -p, --pidfile=[PIDFILE]    Change default location of pidfile
  -P, --port=[port]          Change the port to listen on in server mode
  -q, --no-version           Do not display version in banner
  -t, --timeout=[TIMEOUT]    Set default idle timeout
  -T, --max-timeout          Reset maximum value of timeout allowed
  -u, --umask                 Set default umask(base 8)
      --help                 Print this message
  -V, --version              Print version
  -a, --auth=[AUTH]          Use AUTH for authentication, it can be:
                              default      passwd authentication.
```

To change the behaviour of ftpd-topfield you need to edit the startup script.

To restart ftpd-topfield with your new options just run the start up script, which with the details as above, would be the command:

```
/opt/etc/init.d/S67ftpd-topfield
```

Turbo Mode

The USB link to the topfield can be operated in turbo mode. This gives the fastest file transfer however playback is interrupted and the remote control does not function. The default turbo state is controlled by the the `-b` (or alternatively `-turbo`) option in the startup script `/opt/etc/init.d/S67ftpd-topfield`. If this is included on the command in the start up script then turbo is on by default, otherwise it is off. The default start up script setting is turbo off.

If you change this option, restart ftpd-topfield:

```
/opt/etc/init.d/S67ftpd-topfield
```

If your ftp client allows raw ftp commands like filezilla then you can change the turbo setting on the fly. The ftp command is one of:

```
site turbo
site turbo 1
site turbo 0
```

The first toggles the turbo state so if it was off it becomes on and vice versa. The second turns turbo on in all cases and the last turbo off in all cases. For all 3 the ftp response window reports the new state of turbo.

Note that when you connect to the Asus a new version of ftpd-topfield is started so each time the starting state of turbo is as set in the start up script, not how you last left it when you logged off the ftp session.

FTP clients

Windows has a built-in command-prompt ftp client. Just type

```
ftp 192.168.1.1
```

(use the IP address of your Asus if you have changed it from the default).

You will be prompted for a user name and password, use anonymous, and blank for the password.

Filezilla

Filezilla (<http://filezilla.sourceforge.net/>) is an open-source ftp client that is easy to use with the Asus.

You just need to put the IP address for the ASUS in (ie 192.168.1.151), leave the port as the default and logontype as Anonymous. If you click connect you should get the directory structure of the Toppy in the right hand plane.

Novell NetDrive

Novell NetDrive is an FTP client that allows you to map FTP sites to a Windows drive letter, so you can browse your Toppy's files using Windows Explorer (it also supports the WebDAV protocol).

It is very easy to set up, just install, fill in one page of options and it just works. The URL is your Asus IP address, server type is FTP and it is an anonymous login.

It's available from a variety of sources, just search for 'Novell Netdrive'.

Windows file sharing access to the Asus using Samba

With ftpd-topfield installed, you now have ftp access to files on the Toppo, but no mechanism for transferring files to and from the Asus – this is necessary if you want to install rt2mei.

Samba can be used for this – it basically shares directories using the same protocol as Windows File and Printer Sharing (the SMB protocol, using NetBIOS over TCP/IP, also known as NBT).

This allows you to browse your Network Places in Windows Explorer to navigate through the Asus filing system, and you can map the Asus shared drives as Windows drives.

Important: if you are going to use Samba to transfer script files created in Windows to the Asus, you will have problems when you come to run them, as Linux and Windows use different characters to indicate the end of a line. You need to make sure you use an editor that can be set to use Unix-style line breaks. One option is [EditPlus](#). If you use EditPlus, check under Document | File Format (CR/LF) and make sure this is set to UNIX for every script file you edit on the PC.

Samba comes preinstalled as part of Oleg's firmware, but the web interface is not comprehensive enough to configure it properly.

There is a macsat guide at <http://www.macsat.com/macsat/content/view/27/29/> covering Samba configuration, which formed the basis of my configuration. However the configuration in this guide is a little different. The log files are stored in the /tmp directory structure (which is based in RAM) and the startup script contains additional lines to stop any previously-running instances (which means the script can be repeatedly called to restart Samba, for instance if you change the configuration file).

One important point here is that you must set a Host Name in the web configuration interface – go to the LAN section under IP Config if you haven't done this already, enter a Host Name, and don't forget to Apply, Finish and Save & Restart.

Set up the Samba configuration file

Create a directory for the Samba configuration file, and create the file

```
mkdir /opt/etc/samba
touch /opt/etc/samba/smb.conf
```

Edit the configuration file to be as follows:

/opt/etc/samba/smb.conf

```
[global]
workgroup = WORKGROUP
guest account = nobody
security = share
browseable = yes
guest ok = yes
log level = 1
max log size = 100
encrypt passwords = yes
dns proxy = no

[opt]
path = /opt
writeable = yes
browseable = yes
force user = admin
```

Replace WORKGROUP with the name of the workgroup you are using on the windows network.

You can add as many additional shares as you like, following the structure of the [opt] section. Note that the section name (opt) is the folder name as it will be seen by your Windows PC, and this is mapped to the Asus folder defined by the path variable.

For instance, if you wanted to share the Asus /tmp folder, which is where logs are stored, and where the rt2mei output is stored (see later) you would add to the bottom of smb.conf:

```
[tmp]
path = /tmp
writeable = yes
browseable = yes
force user = admin
```

Set up the Samba start-on-boot script

Create a Samba startup script in `/opt/etc/init.d`, so that Samba will be started automatically when you reboot the Asus:

`/opt/etc/init.d/S97Samba`

```
#!/bin/sh

# get rid of any already-running instances
killall smbd
killall nmbd

# start samba
/usr/sbin/smbd -D -l /tmp/var/log/smb -s /opt/etc/samba/smb.conf
/usr/sbin/nmbd -D -n WL500gP -o -l /tmp/var/log/nmb -s
/opt/etc/samba/smb.conf
```

Replace `WL500gp` in the `/usr/sbin/nmbd` line with the name of your Asus. This is the Windows file-sharing name of your Asus, we suggest for consistency you use the same name as the host name of your Asus (though they don't have to be the same).

Now make this script executable:

```
chmod +x /opt/etc/init.d/S97Samba
```

You can now reboot the Asus. When it has restarted, check that Samba is running using

```
ps | grep mbd
```

If Samba is running, then you should see three lines of output, two of which identify `smbd` and `nmbd`.

Command-line access to the Toppo using Puppy

Puppy provides command-line access to the Toppo from the Asus itself (but not from your PC). It is needed by the automated `rt2mei` EPG download process so that the EPG can be copied to the Toppo.

Install it using `ipkg`:

```
ipkg install puppy
```

Refer to the puppy documentation for more information.

Set up cron for scheduling tasks

`cron` allows scripts to be scheduled and run at regular intervals. Oleg's firmware includes `cron`.

It is also possible to install `cron` using `ipkg`. If you do this, the instructions below won't work (and nor will the `rt2mei` automation instructions). If you want to check if you have the `ipkg` version of `cron` installed, type

```
ipkg list_installed | grep cron
```

If you see the `cron` package listed, you can remove it using the following command but **please note** that any existing `cron` jobs you have set up will be lost, so you will have to add them again.

```
ipkg remove cron
```

To start cron, and configure it to run from files on the USB key¹ it's necessary to create a startup script in /opt/etc/init.d, and to create a directory on the USB key to store the files.

So, create the directory with

```
mkdir -p /opt/var/cron/crontabs
```

And then create the startup script as:

/opt/etc/init.d/S10cron

```
#!/bin/sh
#
# Startup script for cron
#
# use built-in cron, which expects
# to use /var/spool/cron/crontabs for
# crontabs, but symlink /var/spool/cron
# to /opt/var/cron, so changes stored on
# USB storage

# first check /var/spool is present, if not, create
if [ ! -d /var/spool ]
then
    mkdir /var/spool
fi

# check if /var/spool/cron present, if not,
# symlink to /opt/var/cron

if [ ! -d /var/spool/cron ]
then
    ln -s /opt/var/cron /var/spool/cron
fi

# Stop myself if running
killall crond

# start (built-in) cron
crond
```

Now make the script executable:

```
chmod +x /opt/etc/init.d/S10cron
```

You can either reboot to start cron, or execute this script manually with

```
/opt/etc/init.d/S10cron
```

¹ By default, crond and crontab expect to use /var/spool/cron/crontabs for the crontab location. As this is a filing system in RAM, rebooting would loose changes, or would involve flashfs save && flashfs commit every time a change is involved. Therefore the configuration in this guide is to create a symbolic link from /var/spool/cron to /opt/var/cron on the USB key.

Set up automatic switching between GMT and BST

This describes how to set up the timezone on the Asus so that it correctly compensates for changes between summer and winter time.

Without this, you will find your cron jobs jump ahead (or back) an hour when the changeover occurs.

To do this, in a telnet session type:

```
nvramp set time_zone=GMT+0BST-1,M3.5.0/01:00:00,M10.5.0/02:00:00
nvramp commit
```

Note the `nvramp commit` to save this – this information is not saved on the USB key, which is why you need to issue this command.

Adding cron jobs

To add and edit scheduled jobs, type

```
crontab -e
```

This launches the vi editor to edit your existing crontabs.

If you would prefer to use nano, type

```
export VISUAL=nano
crontab -e
```

You need to type the `export` command every time – if you want to make nano default editor (so you don't have to type the `export` command) then see Setting up a `.profile` file to store the default crontab editor on page 33.

Add your new job at the end. Refer to documentation on cron and crontab for the syntax for these lines, for example <http://www.adminschoice.com/docs/crontab.htm>.

Automated download of Radio Times EPG using rt2mei

This section covers the installation of wooder's `rt2mei`, which is a php program to download the Radio Times EPG and transform it into a suitable format for the MyStuff TAP, and also includes how to automate this so it runs once a day.

The NSLU2 package is used, as it contains the automation script, from <http://www.wooders.co.uk/rt2mei/nslu2.php>, follow the download link to download the latest version of the package.

Install prerequisites and configure

Make sure you have installed puppy and have set up cron according to the previous sections.

Install php, `libuclibc++` and `libiconv` (required by php).

```
ipkg install php
ipkg install libuclibc++
ipkg install libiconv
```

Php requires some initial configuration to set the timezone. Edit the `php.ini` file so that the top two lines match this (insert the `date.timezone` line beneath the `[PHP]` line – in fact the `date.timezone` line can go anywhere in the `[PHP]` section)

/opt/etc/php.ini

```
[PHP]
date.timezone = "Europe/London"
```

Install, configure and test rt2mei

Rt2mei will be installed at /opt/usr/local/rt2mei, with output files and logs stored at /tmp/var/rt2mei.

Create the directory for rt2mei:

```
mkdir -p /opt/usr/local/rt2mei
```

You now need to transfer files from the rt2mei package you downloaded to the Asus. The easiest way to do this is if you have installed Samba, you can then use Windows Explorer to map a drive to the Asus opt share.

From the rt2mei package you downloaded, locate the files rt2mei.cfg and rt2mei.php, and copy them to /opt/usr/local/rt2mei

Configure rt2mei by editing /opt/usr/local/rt2mei.cfg, refer to <http://www.wooders.co.uk/rt2mei/configuration.php> for more details on this.

Test that it works:

```
cd /opt/usr/local/rt2mei
php rt2mei.php
```

This will take a few minutes to execute, you'll be able to see the progress in your telnet window. Once it has finished you should see output files in this directory (with ls). Once you are happy with your configuration and that all is working, remove these output files (using rm 'filename'). When you've set up the automated version /tmp will be used for these files.

Install and test the automation

Copy the automation shell script to /opt/usr/local/rt2mei/ -- this is in the package you downloaded, and is named rt2mei.

Make the script executable

```
chmod +x /opt/usr/local/rt2mei/rt2mei
```

Test the script, first creating the /tmp directory for the output:

```
mkdir /tmp/var/rt2mei
/opt/usr/local/rt2mei/rt2mei -new -tmp /tmp/var/rt2mei -ramdisk
```

Again this will take several minutes to complete.

Check the files under /tmp/var for correct output (you might like to add a Samba share for /tmp, see the Samba section for more details).

Now we need to set up a calling script for this, that will also create the /tmp directory automatically:

/opt/usr/local/rt2mei/do-rt2mei.sh

```
#!/bin/sh
#
# rt2mei calling script
#
# make sure /tmp directory exists
# then call the rt2mei script

if [ ! -d /tmp/var/rt2mei ]
then
    mkdir -p /tmp/var/rt2mei
fi

/opt/usr/local/rt2mei/rt2mei -new -tmp /tmp/var/rt2mei -ramdisk
```

Make this script executable with

```
chmod +x /opt/usr/local/rt2mei/do-rt2mei.sh
```

Now you need to set up the scheduled cron job to run the rt2mei process daily.

We do this using crontab (note, this assumes you don't have any other cron jobs set up, if you do they will be deleted).

To do this, type

```
echo "5 7 * * * /opt/usr/local/rt2mei/do-rt2mei.sh >/tmp/var/log/rt2mei
2>&1" | crontab -
```

If you have other crontab entries already set up (you can list them with crontab -l) then you can use

```
crontab -e
```

to launch the vi editor (or see Adding cron jobs on page 26 for how to do this with nano), and add the entry

```
5 7 * * * /opt/usr/local/rt2mei/do-rt2mei.sh >/tmp/var/log/rt2mei 2>&1
```

manually. A log file of the last run will be kept in /tmp/var/log/rt2mei – if you want to keep more than the last run, you can change > to >> in the crontab entry, but note that this will result in ever-growing log files.

If you want to test this first, you can use different time entries.

- a) the "5" corresponds to minutes-past-the-hour
- b) the "7" corresponds to the hour (using 24hour time)

If you do this, choose a time at least 5 minutes ahead of the current time. You can find the current time (and date) on the Asus by typing

```
date
```

Remember to reissue the echo command above with the correct time in it once you've finished testing.

You might also want to add a Toppo wake-up timer, set for 10 minutes or so prior to the crontab time of 7:05am, with a duration of an hour or so, to ensure the Toppo is on when the rt2mei update occurs. Details on how to set wake-up timers are included in the MyStuff manual.

Secure access using ssh: external access to your Asus

Oleg's firmware includes the dropbear sshd daemon.

This allows you to start a terminal session with the Asus using ssh instead of telnet for command-line access.

ssh also allows "tunnelling".

Tunnelling means that network traffic of a particular protocol (eg http, port 80), is "tunnelled" by the ssh client on the PC, and is actually sent over port 22 using the secure ssh protocol to the Asus. dropbear on the Asus then un-tunnels the traffic (ie converting it back to http, port 80 traffic) and sends it on to the recipient application (ie the web server). The same process occurs when the recipient application (in this case a web server) responds – the response is also tunnelled through port 22. As all traffic is sent via ssh, this means it is securely encrypted.

There are many different configuration options for ssh tunnelling, and you are recommended to read a guide such as

http://www.ssh.com/support/documentation/online/ssh/winhelp/32/Tunneling_Explained.html for more information.

The particular configuration we will use here is:

- a) An http server is running on the Asus on the standard http port 80 (this is the normal web administration interface of your Asus)
- b) We wish to connect from a PC (possibly any PC connected to the internet)
- c) The PC will be configured so the ssh client forwards localhost (ie traffic addressed to the PC) on port 1234 (we use 1234 just in case you are also running a web server on the PC) through ssh

So when you type the address localhost:1234 in your browser, the traffic is forwarded to http port 80 on the Asus, but it is tunnelled through ssh, forming a secure connection. The same principle can be used for other protocols.

As ssh is highly secure, you can then set up port/address mappings in your main ADSL modem/router to allow access from the outside world.

You would configure your ADSL modem/router to forward all traffic on port 22 to the IP address of your Asus. And if you were trying to access your Asus from an internet-connected PC, you would use the public IP address of your main ADSL modem/router.

IMPORTANT SECURITY NOTE:

If you set up a port mapping in your ADSL modem/router to give external access, **make sure** you change your password to a strong one. Otherwise it won't be long before someone has hacked into your Asus. And remember if someone does manage to hack in with ssh, they effectively could gain access to your whole network.

dropbear configuration

The dropbear configuration in this guide has the key files stored in /opt/etc/dropbear, and there is an init.d script to start it automatically.

Note that for ssh tunnelling, the configuration is all done by the client, you do not need to set up anything on the Asus for tunnelling to happen.

Create dropbear security keys

To set up dropbear, you first need to create a directory for the key files, and then need to create these keys:

```
mkdir /opt/etc/dropbear
dropbearkey -t dss -f /opt/etc/dropbear/dropbear_dss_host_key
dropbearkey -t rsa -f /opt/etc/dropbear/dropbear_rsa_host_key
```


Create dropbear start-on-boot script

Then create the script to start dropbear on boot:

```
/opt/etc/init.d/S51dropbear
```

```
#!/bin/sh
#
# S51dropbear
#
# start dropbear

# kill already running instances
killall dropbear

# start, using config from /opt/etc/dropbear
dropbear -d /opt/etc/dropbear/dropbear_dss_host_key -r
/opt/etc/dropbear/dropbear_rsa_host_key
```

Then make this script executable with

```
chmod +x /opt/etc/init.d/S51dropbear
```

You'll need to reboot the Asus, or run this script manually, to start dropbear.

Starting a terminal session using ssh: PuTTY

This allows you to do the same as you would normally do in a telnet session, but through ssh, for added security.

If you are using PuTTY, then

- For Host Name, enter the IP address of the Asus
- For Protocol, select SSH
- Click Open to connect

You will be prompted to save the server's key fingerprint in the registry. Select "Yes" here, and you won't get this prompt the next time you connect. Or select "No" and you'll get the prompt every time.

As with a Telnet session, you'll be prompted for your user ID and password.

Starting a terminal session using ssh: Cygwin/OpenSSH

If you are using command-line ssh (eg you have Cygwin and OpenSSH installed), then simply type.

```
ssh admin@192.168.1.1
```

Change the IP address to match that of your Asus, and if you're not using the default admin user ID change that as well. If you are trying to connect through a remote internet-connected PC, replace the IP address with the public IP address of your ADSL modem/router.

You'll be prompted to log in in the normal way. You also may be asked whether you trust the server, answer yes to this.

Using ssh tunnelling to connect to your Asus: PuTTY

The following configuration will map the localhost: 1234 address to port 80 on your Asus, ie http, through a secure ssh tunnel.

- For Host Name, put the IP address of your Asus. If you are connecting through a remote, internet-connected PC, put the IP address of your ADSL modem/router
- Select SSH for the protocol
- Click on "Tunnels" under Connection -> SSH
- Under Source Port, put 1234
- Under Destination, put the IP address of your Asus, and the port to forward, eg 192.168.1.1:80 (use the IP address of the Asus on your local network for 192.168.1.1)

- f) Click on Add
- g) Click on Open to connect

Log on in the normal way in the terminal window.

You can now point your browser at localhost:1234 to access the web administration interface of your Asus.

You can use the PuTTY window as a normal terminal session in the mean time.

Once you have finished, type exit to close the session.

Using ssh tunnelling to connect to your Asus: Cygwin/OpenSSH

The following command on your client PC will cause traffic addressed to localhost:1234 to be forwarded through ssh to your Asus on port 80 (ie http).

```
ssh admin@192.168.1.1 -L 1234:192.168.1.1:80
```

Note that if you are connecting from a remote PC over the internet:

- a) The first IP address (part of admin@192.168.1.1) should be replaced by the public IP address of your main ADSL modem/router
- b) The second IP address (part of 1234:192.168.1.1:80) should be the IP address of your Asus on your home network.

If you now type localhost:1234 in your browser, you will see the web configuration interface of your Asus.

You can also use the ssh command prompt window as a normal ssh terminal session.

Once you've finished, type exit in the ssh command prompt window to shut down the session and disconnect.

Firewalls and SSH

You may have issues connecting using ssh due to firewalls on the network you are trying to connect from. Some networks only allow a few protocols, such as HTTP, HTTPS, FTP, POP3 and IMAP, and won't permit any ssh traffic on port 22. If that's the case try changing the port mapping configuration in your main ADSL modem/router to map a different external port number (one that your firewall will let through) to port 22 on your Asus. If your main ADSL modem/router won't allow mappings which use different internal and external port options, you can modify the `S51dropbear` script and add a port option, eg `-p 8000` would make dropbear use port 8000 instead of port 22.

Sending emails from your Asus

Macsat has a guide for installing esmtp, which is an email client, at <http://www.macsat.com/macsat/content/view/26/29/>.

The installation is very straightforward.

Instead of using the macsat instructions for creating a config file, you can instead use a text editor (nano or vi for example) to create one, an example one is:

```
/opt/etc/esmtprc
```

```
hostname = smtp.yourisp.com:25
username = "yourname"
password = "yourpassword"
```

Replace the values with the ones appropriate to your ISP's mailserver. If your ISP doesn't require a username and password (many don't) then just omit these lines.

Note that the macsat instructions quote the value of the hostname – this is not necessary.

Also note the :25 at the end of the mail server address. Again the macsat instructions don't mention this. You do need to include this value. 25 is normal value for smtp, check your ISP mail server information if you are not sure.

You can test this with

```
echo -e "\nTest Message" | sendmail recipient@example.org -f
sender@example.org
```

Replace `recipient@example.org` with the email address where you want to send the message to, and replace `sender@example.org` with the address the email is being sent from (the reply address). Note the `-e` and the `\n` in the text, this makes sure there is a blank line before the body of the message, which is necessary so the email headers are picked up correctly.

You can then use this in shell scripts to automatically notify you of things happening on your Asus.

For example, here is an `init.d` script to email the post-boot log file whenever the Asus boots up. The filename starts with `S99` so that it is the last `init.d` script to run. Note that we also set one of the email header fields – the subject – at the start, and also that this includes line breaks to indicate where the body of the message starts (`\n`).

/opt/etc/init.d/S99Email-post-boot-log.sh

```
#!/bin/sh
#
# S99Email-post-boot-log.sh
#
# Email myself the post-boot log file
#

message=/tmp/boot-message.txt

# create a file for the email message
echo -e "Subject: Asus has rebooted\n" > $message
echo "/tmp/var/log/post-boot contents follows" >> $message
echo "===START===" >> $message
cat /tmp/var/log/post-boot >> $message
echo "====END====" >> $message

# send it
cat $message | /opt/sbin/sendmail recipient@example.org -f
sender@example.org
```

Replace `recipient@example.org` with where you want the email sent to, and change `sender@example.org` to a suitable reply address (probably both will be your own email address).

Don't forget to make the script executable, or it won't run:

```
chmod +x /opt/etc/init.d/S99Email-post-boot-log.sh
```

If you're using `rt2mei`, you could amend the `/opt/usr/local/rt2mei/do-rt2mei.sh` script to email yourself the `rt2mei` logs every time an update is done.

Other hints and tips

Shutting down and rebooting your Asus

You can shut down your Asus using telnet interface by typing

```
halt
```

You can also shut it down from the box itself by pressing and holding in the Restore button until the power light goes off.

In both cases it should only take a few seconds to shut down – this is indicated by all the lights going off.

You can reboot your Asus using telnet by typing

```
reboot
```

Disabling services from starting on boot

The directory `/opt/etc/init.d` contains all the scripts to run services such as `ftpd-topfield` and `samba` when the Asus boots up.

Only scripts starting with a capital letter S are run, and these are run in order of the two digits following the S in the filename.

To disable a service, eg `ftpd-topfield`, rename the script:

```
mv /opt/etc/init.d/S67ftpd-topfield /opt/etc/init.d/T67ftpd-topfield
```

The `ftpd-topfield` service will no longer start up the next time the Asus is booted.

To enable it again, rename the file back:

```
mv /opt/etc/init.d/T67ftpd-topfield /opt/etc/init.d/S67ftpd-topfield
```

Upgrading the Toppo's firmware

You can upgrade your Toppo's firmware using an ftp client and `ftpd-topfield` running on the Asus.

Simply drop the `.tfd` firmware file into the Toppo's firmware directory.

The Toppo automatically recognises the new firmware file, reboots and loads it.

Setting up a .profile file to store the default crontab editor

Setting up a `/tmp/local/root/.profile` file means you can store common settings that will get recreated when you reboot.

The example here is to store the default editor for `crontab`, saved in the environment variable `VISUAL`.

Create the `.profile` file as (note the filename starts with a dot):

```
/tmp/local/root/.profile
```

```
export VISUAL=nano
```

As this is part of the flashfs filing system, we need to add it to the list of files to be saved, so type

```
echo /tmp/local/root/.profile >> /usr/local/.files
```

And save to make the change permanent with

```
flashfs save
flashfs commit
flashfs enable
```

You can test this by rebooting, then using `crontab -e` to see if `nano` is started by default.

If you want to change your `.profile` file, you can do that whenever you want, but remember that you will need to save it with `flashfs` to make the changes permanent.

Backing up the files on the USB key

Although it's possible to use Samba to share the /opt folder and then to copy all of the files to your PC, if you will do this you will lose some of the Linux file attributes and permissions, so restoring the files won't be that straightforward.

An alternative is to create a backup file on the Asus, and then copy that file to your PC:

```
cd /opt
tar -czf USB-key-backup.tar.gz .
```

Note the trailing full-stop!

You can then copy /opt/USB-key-backup.tar.z to your PC.

To restore the backup, copy your backup file back to the /opt folder, and

```
tar -zxf USB-key-backup.tar.gz
```

Topsy disk space reporting by email

If you have set up esmtp email support, you can set up a cron job to send you a daily email summary of disk space usage on your Asus.

First, create a folder /opt/usr/local/toppyspace, then in that folder, create a file toppyspace.sh, with the following contents:

```
#!/bin/sh
#
# toppyspace.sh
#
# email details of toppy disk usage
#

message=/tmp/toppyspace-message.txt

# header
echo -e "Subject: Topsy disk space usage\n" > $message
# preamble
echo -e "Topsy disk space usage report\n" >> $message

# get disk usage
/opt/bin/puppy -c size >> $message

# send it
cat $message | /opt/sbin/sendmail recipient@example.org -f
                        sender@example.org
```

Replace the email address in the script with your own.

Make the script executable with

```
chmod +x /opt/usr/local/toppyspace/toppyspace.sh
```

Give the script a quick test with

```
/opt/usr/local/toppyspace/toppyspace.sh
```

We now need to schedule it to run every day – I have mine set to run 5 minutes before my rt2mei scripts as I know the Topsy will be turned on at this point.

Edit your crontab with `crontab -e` and add the following line:

```
0 7 * * * /opt/usr/local/toppyspace/toppyspace.sh >> /tmp/var/log/toppyspace
2>&1
```

FAQ

Here we hope to include answers to the most commonly asked questions on the forums.

The rt2mei scheduled download doesn't seem to be working

First of all, check the log file that the scheduled cron job produces. This is `/tmp/var/log/rt2mei`. Make sure you haven't rebooted your Asus in the meantime, otherwise these files will not be present.

If this contains any errors they might point to the source of the problem. If it is not present, it suggests a problem either with your cron installation, or with your crontab. Double-check against the instructions in the guide.

You can step through the rt2mei process manually to see if that gives any clues:

- 1) Try running the rt2mei download process manually from telnet (this process does not transfer files to the Toppo, it just downloads them to the Asus).

```
cd /opt/usr/local/rt2mei
php rt2mei.php
```

If there are any errors, it may be a problem with your `rt2mei.cfg` file

- 2) Try running the full download and transfer to Toppo process with

```
mkdir /tmp/var/rt2mei
/opt/usr/local/rt2mei/rt2mei -new -tmp /tmp/var/rt2mei -ramdisk
```

The `mkdir` command will give you an error if the `/tmp` directory already exists – you can safely ignore that.

Watch for any other error messages.

Once it is complete, check the output files `/tmp/var/rt2mei/My Stuff.mei` and `/tmp/var/rt2mei/My Favourites.mcl` – if you set up the custom Samba configuration and created a share for `/tmp` you can examine these files on your PC.

Now check the files on the Toppo, using ftp (assuming you have `ftpd-topfield` running on the Asus). Look in the `ProgramFiles` directory for `My Stuff.mei` and `My Favourites.mcl`.

- 3) If all of that looks fine, check the EPG on the Toppo. If that doesn't reflect the downloaded EPG files, it suggests a problem with your EPG (eg `MyStuff`) configuration, so check that.

What is stored on the USB key, and what is stored on the Asus?

The majority of the instructions in this guide make changes to files stored on the USB key.

The exceptions are

- a) any configuration changes you make in the web interface
- b) the changes made under cron setup to set up the Asus for daylight saving
- c) the `/usr/local/sbin/post-boot` file
- d) a `/tmp/local/root/.profile` file (if you have created one)

You can make a backup of all configuration changes you make in the web interface, and the daylight saving changes, from the web interface. Go to System Setup -> Settings Management, and follow the instructions there (this page also tells you to how to restore a saved configuration file)

You can make a backup of the flashfs file system on the Asus itself, which contains the post-boot file and the `.profile` file by going to System Setup -> Flashfs Management and following the instructions there (this page also tells you how to restore a saved flashfs file).

So to back up everything, you need to make a copy of

- a) The configuration file saved from the web interface
- b) The flashfs file saved from the web interface
- c) Everything on the USB key

I'm getting a message "Connection closed by remote host" when using FTP

There's a known timeout problem with ftpd-topfield and DNS timeouts. If you experience this error, try changing the DNS IP address in the web interface to your ISP DNS server, rather than using the IP address of your main router.

Thanks

We would never have got our Asus WL-500gP's working with the Toppo without much help from others.

The toppy.org.uk forums proved an invaluable source of information, together with Jackofall's guide for the Asus WL-500G Deluxe, the macsat guides, wooders' rt2mei package and web pages and the Oleg firmware forums.

Particular thanks go to mdryden, vroomfondle, downswalker, paragon and ubuntu for providing help and information, and for road-testing and providing feedback on some of the forum posts which were a precursor of this guide.

And of course thanks to Nigel Whitfield for setting up, hosting and administering toppy.org.uk.

References

Toppo.org.uk <http://www.toppo.org.uk/> and forums <http://forum.toppo.org.uk/forum/index.php>

Jackofall's guide <http://www.itsallgeek.info/Connecting%20an%20ASUS%20WL-500G%20DELUXE%20to%20your%20wireless%20network.pdf>

macsat guides <http://www.macsat.com/macsat/content/category/3/13/29/>

rt2mei <http://www.wooders.co.uk/rt2mei/index.php>

Oleg firmware <http://oleg.wl500g.info/> and forums <http://wl500g.info/forumdisplay.php?f=61>