

## AutoAP

[English](#) • [Deutsch](#) • [Español](#) • [Français](#) • [Italiano](#) • [???](#) • [Polski](#) • [Português](#) • [??????](#) • [Svenska](#) • [???\(???\)?](#) • [???\(???\)?](#)

AutoAP is a script that continuously scans for open Wi-Fi connections, tests them for validity, and connects to the strongest signal. If the connection is lost, the script scans again and finds the strongest valid signal again, and maintains a continuous connection to the internet in a mobile or portable environment. The script parameters are highly configurable, including ability to configure secure connections.

Typically, you would use this script on your router while in Repeater Mode, Client Mode, or Client-Bridged Mode.

*Before using this guide, it is recommended that you know how to:*

... flash your device

... use the Web Interface

... setup Repeater Mode, Client Mode or Client-Bridged Mode.

## Contents

- 1 Installation
  - ◆ 1.1 Dynamically Download
  - ◆ 1.2 Built-in Image
  - ◆ 1.3 JFFS
  - ◆ 1.4 Samba
- 2 Usage
- 3 NVRAM Variables
- 4 Development Status and Bugs
- 5 Links

## Installation

There are several ways to install AutoAP.

## Dynamically Download

If you would like to use autoap with any image/build (such as the latest v24beta), use the following script. It clears wl\_ssid to dynamically connect to any open AP and then every 30 seconds tries to download stand-alone AutoAP script. Once downloaded, AutoAP starts.

◇ Go to tab "Administration", subtab "Command".

◇ Copy and paste the following into the edit box (make sure to get the "&" at the end):

```
nvramp set wl_ssid=""  
`cd /tmp; until [ 2 = 1 ];do /bin/sleep 30;
```

## AutoAP

```
/usr/bin/wget -Oautoap.sh http://downloads.sourceforge.net/autoap/autoap_090711.sh;  
chmod 777 ./autoap.sh; ./autoap.sh; done` &
```

- ◇ Click the Save Startup button
- ◇ Reboot your router
- ◇ Verification: After a minute or two, go to <http://RouterIP/user/autoap.htm> (where RouterIP is your router's IP address) to view the html log, which is setup by default. If you see a log, this indicates Autoap was successfully downloaded and instantiated. You may also telnet into the router and perform a "ps" to verify the script is running.

## Built-in Image

If you would like to have autoap built into an image:

- ◇ Download firmware image (link? url?) with AutoAP script built in (see sourceforge link and caveats in "Links" section below)
- ◇ Flash your router with this firmware
- ◇ After flashing go to tab "Administration", subtab "Command"
- ◇ Copy and paste the following into the edit box:

```
/bin/autoap &
```

- ◇ Click the Save Startup button
- ◇ Reboot your router

## JFFS

You can use JFFS to directly call the AutoAP script.

- ◇ Make sure JFFS is enabled (disabled by default)
- ◇ Make sure SSH is enabled (disabled by default)
- ◇ Download the stand-alone AutoAP script (see Links below)
- ◇ If downloaded file is named something different, rename it to `autoap.sh` and put it into `/jffs/` using SSH.
- ◇ Chmod `autoap.sh` to `777` using SSH (or telnet).
- ◇ Use the same startup script as in "Built-in Image" above but use:

```
/jffs/autoap.sh &
```

- ◇ Instead of:

```
/bin/autoap &
```

- ◇ Reboot router
- ◇ After you have it working, SSH can be disabled.

## Samba

You can use [Samba](#) to directly call the AutoAP script.

- ◇ Obtain the stand-alone AutoAP script (see Links below)
- ◇ Put it somewhere on your local network (using [Samba](#)).
- ◇ Use the same startup script as in "Built-in Image" above but reference your Samba path instead of /bin/autoap.

## Usage

You can change the behavior of AutoAP script by setting [NVRAM](#) variables. For example, let's set AutoAP to prefer to connect to a certain, wep-encrypted network (SSID: my\_net" wep-key: abcdef1234 ) before trying others.

- ◇ Go to tab "Administration", subtab "Command".
- ◇ Copy and paste the following into the edit box:

```
nvramp set autoap_prefssid="my_net*key*abcdef1234"
nvramp commit
```

- ◇ Click the Run Commands button
- ◇ Reboot your router

Now it will only try other APs if the preferred one has no internet connectivity. Of course you could also give more than one preferred AP.

## NVRAM Variables

What follows is a complete list of included [NVRAM](#) variables and what they do (with default values). Be aware that these refer to the most recent versions of autoap. Older version may have different variables. To be sure just check in the history of the wiki for the date of your version.

**nvramp set autoap\_logger="html"** [Valid values: syslog, html, /path/to/logfile]

When set to 'syslog', AutoAP will send all logdata to syslog. Depending on your log level settings, you will see more or less data. AutoAP has quite a bit of log data it sends, however if your log level is set to high, the router should only send out important autoap notices, like new connections, disconnections, or errors. If set to low, it will show you debug data.

When set to 'html', the log data is written to a file available via the web interface at `http://RouterIP/user/autoap.htm`. This log is kept trimmed to *autoap\_logsize* lines.

When set to a path, i.e. '/tmp/newautoap.log', the log data will be saved to that location. This will also be trimmed to *autoap\_logsize* lines.

**nvramp set autoap\_findopen="1"**

When set to 1, AutoAP will search for unencrypted, open access points. When set to 0, it will not connect to any open access points.

## AutoAP

### **nvramp set autoap\_findwep="0"**

When set to 1, AutoAP will search for WEP encrypted access points. If you set this to one, make sure to add keys in the next settings. Set to 0 keeps AutoAP from trying to connect with any WEP networks. This is far too slow to use as a brute force tool, so if you don't know the WEP keys to networks you're trying to connect with, just keep this at 0.

### **nvramp set autoap\_wepkeys=""**

This is a space separated list of WEP keys you want AutoAP to use when trying to connect to WEP networks. For instance, if you set "AAABBBCCC1 4455993322 111122222" it will attempt to use those three keys, in that order, to connect to any WEP (not WPA) access points.

A space separated list of SSIDs with WEP keys you want AutoAP to use when trying to connect to WEP networks. Spaces in SSIDs have to be replaced by '\*'. Example: "My Home" -> "My\*Home". The key is appended directly, preceded by a "\*key\*". Example: "My Home", WEP key:1234567ab -> "My\*Home\*key\*1234567ab SSID2\*key\*ab1212cd"  
To use these the option autoap\_findwep has to be enabled as well.

### **nvramp set autoap\_aplimit="10"**

The number of AP's that AutoAP should keep track of at one time. The default is 10, so after trying to connect to 10 access points, AutoAP will take another full scan and get a fresh list of URLs. If you live in a Wi-Fi heavy area, and your router doesn't move much, you can safely increase this to 15 or so. If you are in a sparse area, or routers around you tend to have weak signals, set it lower.

Also, if you're moving (i.e. router is in a car) you'll want this number to be somewhere between 2 and 5.

### **nvramp set autoap\_inet="1"**

When set to 1, it will check to make sure internet is active on an AP before considering it valid. When set to 0, it only checks for connection to the gateway.

### **nvramp set autoap\_ineturl="www.google.com"**

The URL to check.

### **nvramp set autoap\_dhcpw="15"**

The amount of time to wait for a DHCP renewal to complete. If the routers you're accessing are fairly quick, you can decrease this to ~8-10 and save some time on the scans. However if the signal is low quality, or the router is slow to respond, you can increase it. The default should be a safe bet for most.

### **nvramp set autoap\_scanfreq="60"**

The amount of time to wait between scans. When AutoAP finds a valid connection, it checks every so often to ensure that the connection is still active. If it isn't, it seeks out a new AP.

This number controls how often it should check.

This only applies to the frequency of scans when the connection is good. When there is no connection, or no APs, it will automatically scan more frequently.

### **nvramp set autoap\_macfilter=""**

A space separated list of MAC addresses in the format xx:xx:xx:xx:xx:xx that AutoAP should avoid. If you have routers that you don't want to log on to, add them here. The MAC filter is more accurate than the SSID filter, as no two routers share a MAC address. (Unless cloned I suppose).

### **nvramp set autoap\_ssidgefilter=""**

A space separated list of SSIDs to avoid. Normally the macfilter will be a safer bet, but occasionally there may be a WDS network, etc, that you don't want to connect to that includes many APs with the same SSID. Spaces in SSIDs have to be replaced by '\*'. Example: "My Home" -> "My\*Home".

### **nvramp set autoap\_logsize="1000"**

## AutoAP

The maximum number of lines the logfiles will contain (for html and path based logging only). Once the size of the log reaches this value, it will be trimmed down to 3/4th of it, keeping the most recent entries. You may want to adapt this value depending on the memory of your router.

**nvramp set autoap\_prefssid=""**

A space separated list of SSIDs to try to connect to first. Spaces in SSIDs have to be replaced by '\*'. Example: "My Home" -> "My\*Home".

If a preferred network is WEP encrypted, append the key directly, preceded by a "\*key\*".

Example: "My Home", WEP key:1234567ab -> "My\*Home\*key\*1234567ab"

**nvramp set autoap\_prefonly="0"**

If "1", exclusively try to connect to preferred networks.

**nvramp set autoap\_rescandelay="120"**

Number of 'autoap\_scanfreq' after which the nvramp variables are reread, and a rescan is forced, even if currently connected. This makes sure that you are connected to your preferred networks even if at some point they were unavailable, and it also takes advantage of new APs. Setting this to a large number practically deactivates the feature (3600 is about 1 day).

**nvramp set autoap\_refreshnow="0"**

If "1", during the next 'autoap\_scanfreq' force a reloading of nvramp variables and rescan for networks. This is handy if you modify a nvramp variable and you don't want to wait to the next refreshdelay to take it into account. After recanning it is obviously automatically reset to '0'

## Development Status and Bugs

AutoAP has an active project page on SourceForge.net [1], where bugs are collected and development updates are made. It is important that you submit bugs to the sourceforge project page should you encounter any, as that is the only way we can be sure that the problem is addressed.

You can also download the standalone version of AutoAP, sync to svn and access documentation as it becomes more prevalent.

## Links

- ◇ [Project AutoAP](#) on SourceForge site where you can download the stand-alone AutoAP script, or firmwares with AutoAP script built in.
- ◇ Current development version of the stand-alone [script on sourceforge](#) (see [Changelog](#)). There is no sense in continuously updating the "latest version" information here, so please check the Sourceforge website for the latest version of the script.
- ◇ The AutoAP [discussion thread](#)