For a Linux installation guide on Kismet see: wrt54g kismet with linux server

Kismet is a layer 2 wireless network detector, sniffer, and intrusion detection kit.

Kismet is an OSS project for Linux. Find out more information about it at the Official Kismet Wireless site.

Since Kismet is a Linux project, it can run as either the full server, or just a drone server on the WRT54G router. You are then free to run a Kismet client on your desktop computer view the output.

Anyone familiar with Network Stumbler will definitely appreciate running Kismet Server or Kismet Drone on their WRT54G, since this router has some of the best radios around, along with nice sturdy RP-TNC connectors to hook up even better antennas without worrying about damaging your wireless card or radio pigtails (I've gone through a number of PCI cards with RP-SMA connectors that keep breaking apart).

Contents

Server or Drone

Drone

The Kismet project has developed the capability of running Drones: devices with wireless cards that merely send their data back to a Kismet server. If you run the Kismet Drone on your DD-WRT box, you'll need to run the client and the server elsewhere. Both can be on your desktop computer, or you could have a machine somewhere else running just the server. Running a Kismet drone on your WRT and the Client/Server on a desktop computer(s) seems to be the most common installation for windows users

While drones require a little more work to install, they provide 3 benefits over running full servers on the WRT:

1. It is easier to save captured packets to use later, such as for WEP decryption.
2. Drone installations are smaller, requiring less space on the WRT
3. Drone installations probably require less processing power allowing you to run more additional programs on your WRT.

Install Drone on WRT54G/S

Server

Since the kismet server is capable of running on top of DD-WRT one is able to simplify the installation by only running a client on their desktop machine. I'm not exactly sure what the advantages are, except that this seemed to be easier to install. Since the actual server is running on the DD-WRT, any packets captured will be saved on the WRT, meaning you will have to copy them to your desktop if you plan to interpret them (such as for WEP decryption). This is why running the drone only on the WRT, and running the server+client on the desktop is a better idea as captured packets would be saved to the hard drive desktop machine rather than using the limited RAM on the WRT.
One of the advantages of a server on your desktop machine is that multiple drones connected to the server will only generate one log file. Install Server on WRT54G/S

Installing & Configuring on the WRT54G

Drone

IMPORTANT NOTE: This installation guide describes how to set up BOTH the server and the drone on the WRT. If you want to capture packets the server needs to run on your Linux box (windows might also work but you would need to compile the server within cygwin - I don`t think this works: the last links to another how-to on this site says it won`t compile with the pcap option within cygwin).

Question: What is the point of installing the drone on the WRT? if you're going to put the server there too? Why not just do the server on the WRT? I thought the whole point in putting the drone on the WRT is so you can put the server on your desktop machine *instead* of on the WRT??

Answer: This is exactly the problem - at the moment it is REALLY pretty much useless.
Normally you would install only the server on the WRT to get a fast result (as you said above). The disadvantage is that you can`t capture packets easily - you may do it with the SD Card-Mod, by mounting an SMB or NFS share or by simply copying the files from the WRT but this quite complicated (especially for larger dumps). Having the drone on the WRT would mean that you could run the server on your desktop to log the packets - as I have only got windows I can`t run the server on my box (I didn`t see that I couldn`t log packets this way - it was a little late when I did this...). As a defense I have to say that the other that is linked on this page also uses the drone and the server on the WRT. If you have Linux this how-to might be useful on how to set up an drone - on linux you can dump packets.
I have also found out that because of some weird problem the server alone supports channel hopping - the drone from rops site doesn`t do this (it says that the source (prism0) isn`t able to do it - but the server does it ???). Simply try it - it might work for you.

Work Around for server on windows issues.
With the high availability of Virtualization software, it is rather simple to install a virtual machine onto a windows box. VMWare and Microsoft both offer free virtualization software for windows. Microsoft also offers a server version of their Virtualization software, and VMWare sells an "enterprise" edition of their software which has many additional features. Virtual Box is an open source virtualization server. I have no experience with it, though it's USB support features warrant a look.
After you install your chosen Virtual PC, follow the programs instructions to create a virtual machine. You will also need to assign it hard drive space. It can either use a drive you specify, or you can create a Virtual Hard Drive [VHD] which is simply a file stored on your hard drive that holds the file system and files on your virtual pc. Once you have a VM created, you can pick a linux distribution, follow standard install procedures, and install the kismet server software on your linux VM. From here, you have the choice of installing the windows client on your pc, or the linux client inside the linux VM.
VMware Server Wiki
Virtual PC Wiki
Virtual Box's Wiki
A Handy comparison chart of available VM software

Assumptions

- Your router has SSH configured and you can send files via SSH to and from your desktop

- You have a text editor capable of Unix line delimiters. TextPad and win32pad are both acceptable (and free).
  Failure to use one of these when editing text files will prevent your installation from working
- You have telnet or SSH enabled on your router.
- You have jffs enabled or do mind redoing this whenever the router reboots
- I will assume you are installing to the jffs partition. If you are not, replace jffs with tmp or some other folder

Preparation

WARNING: These packages are rather old (and doesn't work at least for WRT-54GL v1.1) use the ones linked from the article Wrt54g_kismet_with_linux_server

1. Download the Kismet-Drone-Package to your computer.
2. If you're going to run the server. download the kismet-Server-Package too.
3. Now rename them from *.ipk to *.tar.gz and extract them.
   - ♦ When everything is extracted you may delete the control folders since we don`t need them.
4. The following files are needed; copy them to a folder named "kismet":
   - ♦ data/usr/bin/kismet_drone
   - ♦ data/etc/kismet_drone.conf
5. If you're running the server, copy these files as well:
   - ♦ data/usr/bin/kismet_server
   - ♦ data/etc/kismet.conf
6. Edit the kismet_drone.conf file
   1. Find the line "source=wrt54g..." and change to "source=wrt54g,prism0,drone"

Copy the Files

1. Load up WinSCP or another SCP client.
2. Browse to /jfss on your router.
3. -Copy the folder "kismet" from your computer to the /jfss folder on your wrt54g

Installation

1. Telnet/SSH to your router and finish the configuration.
2. Disable AP mode and enable passive mode (alternatively, you can choose client mode from the web interface)
   1. enter the command "set wl ap 0"
   2. enter the command "set wl passive 1"
3. Make the binaries executable.
   1. enter the command "chmod 755 /jffs/kismet/kismet_drone"
   2. If running the server, enter the command "chmod 755 /jffs/kismet/kismet_server"

You may also do this by using WinSCP:

- Open the properties for the files (press "F9") and check the "x" in the line "owner"

Run Kismet

There are three ways run Kismet:

- Launch automatically at server startup (let me know if you know how to do this).
- Launch manually by Telnet or SSH
- Launch manually by Web Interface

Automatically

In principle you would simply need to install the Startup Scripts out of the data directory that came within the *.ipks
- I don't know how to do this. I guess you would also have to edit them - they are using different paths than we used
for the installation of the binaries and configuration files.

Continue to Desktop Configuration

Telnet/SSH

Disadvantage
: Server will stop if you close the telnet/putty window while the drone will continue running.

1. Telnet/SSH into the Router. And enter the following commands:

```
/jffs/kismet/kismet_drone -f /jffs/kismet/kismet_drone.conf
/jffs/kismet/kismet_server -f /jffs/kismet/kismet.conf
```

Continue to Desktop Configuration

Web Interace

Disadvantage
: Server doesn't always start correctly, forcing a reboot to fix.

1. Using the Web Interface goto the **Administration** tab and the **Commands** subtab (older DD-WRT versions have this as **Diagnostics** subtab.
2. Enter the following commands:

```
/jffs/kismet/kismet_drone -f /jffs/kismet/kismet_drone.conf and click cmd
/jffs/kismet/kismet_server -f /jffs/kismet/kismet.conf and click cmd
```

Continue to Desktop Configuration

(If you are just running the drone on the wrt, you will have to use a short script to manually channel hop. (This may also be needed even if you run the server on the wrt) see channel hopping on kismet drone for more details.)

see also installing wrt54g kismet with linux server

Server

Assumptions

- Your router has SSH configured and you can send files via SSH to and from your desktop
- You have a text editor capable of Unix line delimiters. TextPad and win32pad are both acceptable (and free).
  Failure to use one of these when editing text files will prevent your installation from working
- You have telnet enabled on your router.
- You have jffs enabled or do mind redoing this whenever the router reboots
- I will assume you are installing to the jffs partition. If you are not, replace jffs with tmp or some other folder

Preparation

1. Download the muchasta.com binary
2. Extract the files to somewhere on your computer.
3. Edit kismet.conf file
    - ♦ Find the line "source=wrt54g..." and change it to "source=wrt54g,prism0,wrt54g"
    - ♦ Find the line "allowedhosts=...." and change it to "allowedhosts=192.168.1.0/24"
      (Network-Hardware-IPs: 192.168.1.1 -> 192.168.1.254 Subnet-Mask: 255.255.255.0)

NOTE: If you are not using 192.168.1.x for your network, substitute your network setup.

Copy the Files

1. Load up WinSCP or another other SCP client
2. Browse to /jfss on your router and make the folder "kismet_server"
3. Copy kismet_server and kismet.conf from your computer to the /jfss/kismet_server folder on your wrt54g

Installation

1. Telnet or SSH into your router
2. Disable AP mode and enable passive mode (alternatively, you can choose client mode from the web interface)
    - ♦ enter the command "wl ap 0"
    - ♦ enter the command "wl passive 1"
3. Make the server binary executabe
    - ♦ enter the command "chmod 755 /jffs/kismet_server/kismet_server"

Run the Server

There are many ways you can run the server.

- Launch automatically at server startup (let me know if you know how to do this).
- Launch automatically using a client, such as [musatcha.com]
- Launch manually by Telnet or SSH
- Launch manually by Web Interface

Automatically at startup

Continue to Desktop Configuration

Telnet/SSH

Disadvantage
: Server will stop if you close the telnet window

1. Telnet/SSH into your Router
2. Enter the command ""

Continue to Desktop Configuration

Web Interface

Disadvantage
: Server doesn't always start correctly, forcing a reboot to fix

1. Using the Web Interface goto the **Administration** tab and the **Commands** subtab (older DD-WRT versions have this as **Diagnostics** subtab.
2. Enter this command into "/jffs/kismet_server/kismet_server -n -f /jffs/kismet_server/kismet.conf" into the command box.
3. Click Run Commands

Continue to Desktop Configuration

Using musatcha.com client

Disadvantage
: Currently can't send Usernames and Passwords, so you have to launch using another method.

Visit Musatcha's howto and view step 5.

Continue to Desktop Configuration

Installing & Configuring on the Desktop

When installed on WRT as drone

This is simple: use method B or C as if you had set up only the server (maybe A - the whole WiFi mapping software didn`t work for me so I didn`t test it)

When installed on WRT as server

1. Choose a client:
    - ♦ Linux users can Kismet client
    - ♦ Windows users can do any of the following:
        - ◊ Use Musatcha.com's WiFi mapping
        - ◊ Compile Kismet to run on Cygwin
        - ◊ Run the precompiled Kismet on Cygwin
        - ◊ Kiswin (client/server), Joshua Wright's standalone package, supposed to work with only the kismet_drone installed on the WRT54G!

musatcha.com's WiFi Mapping Software

- Read Step 5 from his guide

Compile Kismet to run on Cygwin

1. Install Cygwin, a *nix environment for Windows. .
    - ♦ Ensure that the developer tools are installed. You'll need gcc, make, subversion, etc.
2. SVN the source into your cygwin
3. make, etc. See the guide in the external links. I couldn't get this to work.

Precompiled Kismet for Windows on Cygwin

1. Cygwin, a *nix environment for Windows.
    - ♦ Minimal install is fine.
2. Download and unzip the precompiled client
3. Edit kismet_ui.conf to reflect your correct WRT LAN ip address
    - ♦ find "host=192.168.1.1:2501" and change "192.168.1.1" if that's not your router IP
4. Run kismet_client.exe WHILE your server is running on the router. You may have to launch a cygwin command prompt first. Cygwin behaves kinda weird sometimes.
5. The client is entirely keyboard driven. Press the 'h' key for help. You'll probably want to turn off auto sort right
   away so you can view extended information about individual networks. Press the 's' key to do this.

2nd Way - Drone only

If you own a WRT54GS (with 32 megs of RAM), you have an easier way.

1. Open a Telnet/SSH prompt and enter the following commands:

```
ipkg update
cd /tmp
wget http://www.kismetwireless.net/code/kismet-2006-04-R1-wrt54.tar.gz # your can use latest ve
tar -zxvf kismet-2006-04-R1-wrt54.tar.gz # update if you get another version
cd kismet-2006-04-R1-wrt54 # again, update if you get another version
vi conf/kismet_drone.conf
```

If you don't know vi, search on google to use it. Set your needed values like:

- *allowedhosts=127.0.0.1,192.168.0.0/24*
- *source=wrt54g,prism0,wrt54g* (Watch out to change the uncommented *source* line, not the commented one!)

Now it's time to have fun, enter the following commands into the Telnet/SSH prompt:

```
wl ap 0
wl disassoc
wl passive 1
wl promisc 1
./kismet_drone -f conf/kismet_drone.conf
```

Once you're done, you can save this permanently to your Jffs or Samba server by doing the following:

```
cp kismet_drone /jffs/
cp conf/kismet_drone.conf /jffs/
```

And run kismet_drone this way:

```
cd /jffs/
wl ap 0
wl disassoc
wl passive 1
wl promisc 1
./kismet_drone -f kismet_drone.conf
```

To run the server and the client, you're better to use a VMWare image with a live Linux CD if you are a Windows user and you don't want to install linux on your computer. See also Kismet on Linux.

You can look at http://www.renderlab.net/projects/wrt54g/kismetonwindows.html to compile the latest kismet on your Windows computer.

References

- http://www.renderlab.net/projects/wrt54g/openwrt.html
- http://www5.musatcha.com/musatcha/computers/kismet_on_the_linksys_wap54g.htm
- http://amsterdam-wireless.nl/pipermail/wireless/2005-February/000524.html

- http://www.renderlab.net/projects/wardrive/wrt54g/kismetonwindows.html