Wireless security and Linux

Access Point

If you are looking for a cheap and secure wireless router setup, check out Tomato, DD-WRT, or OpenWrt. *By Kurt Seifried*

I actually remember when I bought my first wireless network card. I was in Vancouver airport, and they were selling them for about US$ 200 with unlimited usage in the airport (as opposed to having to rent one for US$ 20 an hour). At that time, I was spending a lot of time in this airport, so I purchased a card and had a whopping 2Mbps (802.11a) of bandwidth to use while waiting. This purchase was quickly followed by a wireless router so I could enjoy the wireless goodness at home.

Fast forward a decade and now ISPs are giving away wireless N routers like banks used to give away toasters. But what steps have been taken to ensure the security of all these wireless networks? Originally, there was WEP (Wired Equivalent Privacy), which can be broken in real time and is pretty much useless now, then came its successor WPA, which was basically WEP with rotating keys, again pretty useless in practice [1].

Finally, WPA2 came along, which uses the AES encryption algorithm (very strong) and has proper key setup, making it very difficult to break into. And that is pretty much the extent of wireless security for most people. You buy a wireless router, you hopefully set a password on it for the administrative interface (although virtually no wireless routers actually force you to do this), you set up WPA2 with a good password (again none force this), and that's it.

"Open source firmware blows the vendor firmware out of the water."

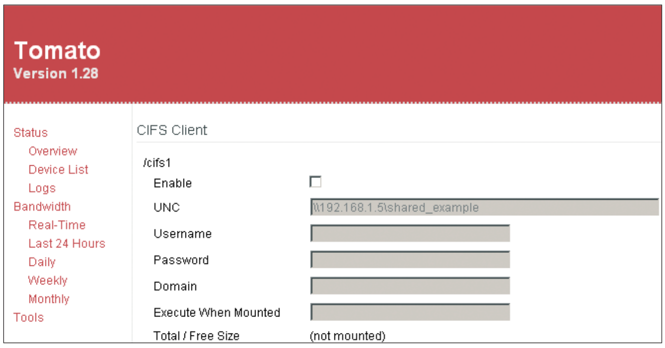
To make matters worse, most of these wireless routers are running pretty minimal operating systems (sometimes referred to as firmware) that have just enough capability to get you online and not much else. Additionally, much of this firmware is either out of date, contains security flaws, or simply does not provide reliable (reset the router daily to keep it working) or fast performance (200Kbps on file transfers over a 15Mb line).

So, what do you do if you want to build a secure router that will support more than just WPA2 and some simple packet passing? What if you want a wireless router that will act as a VPN (e.g., allowing you to bridge access to a corporate network) or to act as a VPN server (e.g., allowing you to connect securely to it from elsewhere on the Internet). Or, what if you need IPv6 support? You have three main options: You can buy a high-end wireless router designed to allow for a more full-featured system (e.g., Mikrotik or Soekris); you can add a wireless card to an existing Linux box and set it up as a wireless router; or you can buy a cheap wireless router that is supported by OpenWrt, DD-WRT, or Tomato.

The first option is pretty simple: You just go spend US$ 100-400 (you buy the system, wireless card(s),

KURT SEIFRIED

Kurt Seifried is an Information Security Consultant specializing in Linux and networks since 1996. He often wonders how it is that technology works on a large scale but often fails on a small scale.



CIFS Client

/cifs1

Enable □

UNC \\192.168.1.5\shared_example

Username

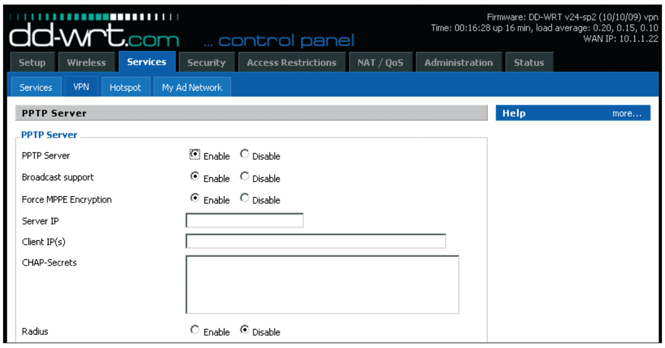
Password

Domain

Execute When Mounted

Total / Free Size (not mounted)

Figure 1: Tomato CIFS client setup.



Firmware: DD-WRT v24-sp2 (10/10/09) vpn
Time: 00:16:28 up 16 min, load average: 0.20, 0.15, 0.10
WAN IP: 10.1.1.22

PPTP Server

PPTP Server

PPTP Server ● Enable ○ Disable

Broadcast support ● Enable ○ Disable

Force MPPE Encryption ● Enable ○ Disable

Server IP

Client IP(s)

CHAP-Secrets

Radius ○ Enable ● Disable

Figure 2: DD-WRT PPTP server setup.

power supply, and enclosure), load up either the vendor-supplied firmware or install a stripped-down system on it, and off you go. The main disadvantages of this are cost, although some really nice enclosures and boards will take three or more wireless cards and provide multiple network interfaces (including Gigabit Ethernet).

The second option is cheaper but faces one problem typically: Firewalls are often hidden away in server rooms, wiring closets, or other areas that are less than ideal for placing aerials. However, if you want to go this route (either because placement isn't a problem or you can run an extension cable for the antenna), then you'll want to check out HostAP for Linux [2].

That brings me to the third option: Buy a cheap wireless router – the advantages are: no moving parts, small, did I mention cheap? – and install custom firmware that provides more capabilities and better reliability and performance. To make things even more interesting, each of the three open source firmware options has a different design philosophy, resulting in three very different products and almost guaranteeing that one will fit your needs.

Tomato

Tomato doesn't include a lot of features but then it isn't meant to; "Tomato is a small, lean and simple replacement firmware" [3], making it the simplest of the three. If you don't need features such as VPN capabilities or network bridging, then Tomato is a great replacement for the vendor-supplied firmware. I really like Tomato's interface. It's incredibly simple, and it's easy to use and configure; setup is a snap as well (Figure 1). If you are simply looking for something more reliable or up to date, this is the one for you.

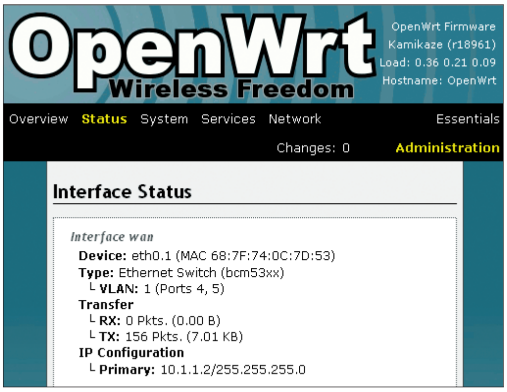
DD-WRT

DD-WRT [4] offers a number of builds, from a Micro and Mini generic with limited capabilities (similar to Tomato) all the way to a VoIP-specific and VPN-specific build. Fortunately, a chart lists all the capabilities and various versions of DD-WRT in the wiki (look for the page called "What_is_DD-WRT"). You have everything from Hotspot, IPv6, OpenVPN, PPTP (see Figure 2), ProFTPD, SNMP, SSH, and Telnetd to a Samba/CIFS client (so you can mount Windows shares onto the device).

I chose the VPN build and would strongly recommend this product if you're looking for good network-related capabilities. It has EoIP (Ethernet over IP, allowing you to bridge networks), VLAN, QoS, and advanced firewalling (including the ability to block specific P2P networks). I also like that it forces a mandatory password change before you can configure it.

OpenWrt

OpenWrt [5] out of the box is pretty minimal, and at first I wasn't too impressed (Figure 3). But then I read about packages. OpenWrt has a package system for additional add-ons, and, boy, do they provide add-ons. It has everything from Squid, NTP, OpenVPN, CUPS (printing support), and lightHTTPD to an IRC server, Nagios (network monitoring), Asterisk (a VoIP server), and the Perl programming language.



OpenWrt Firmware
Kamikaze (r18961)
Load: 0.36 0.21 0.09
Hostname: OpenWrt

Changes: 0

Interface Status

Interface wan

Device: eth0.1 (MAC 68:7F:74:0C:7D:53)
Type: Ethernet Switch (bcm53xx)
└ **VLAN:** 1 (Ports 4, 5)
Transfer
└ **RX:** 0 Pkts. (0.00 B)
└ **TX:** 156 Pkts. (7.01 KB)
IP Configuration
└ **Primary:** 10.1.1.2/255.255.255.0

Figure 3: OpenWrt process control.

Basically, anything you want OpenWrt to do, it can do. The only catch is that you will need a router with a sufficiently large amount of storage space and memory (the WRT54GLs I bought are seriously underpowered, with only 4MB of flash RAM and 16MB of system memory). My advice is to do the research and buy something with 8MB of flash memory (like the WRTSL54GS).

Summary

In every respect, these open source firmware alternatives blow the default vendor-supplied firmware out of the water. Combined with a USB port, you can even have your router do print server or file server duty, or both, for your network, which adds up to a pretty complete package.

If you add OpenWrt's packages into the mix, then it is no contest between OpenWrt and DD-WRT. So, to upgrade your router and make it more secure, I would recommend replacing the default firmware if you can. (Make sure you check the compatibility lists!) ■■■

INFO

[1] Aircrack-ng: *http://www.aircrack-ng.org/*

[2] Host AP: *http://hostap.epitest.fi/*

[3] Tomato: *http://www.polarcloud.com/tomato*

[4] DD-WRT: *http://www.dd-wrt.com/*

[5] OpenWrt: *http://openwrt.org/*