

WIRELESS HACKS™

2nd
Edition

*Tips & Tools for Building, Extending,
and Securing Your Network*



O'REILLY®

Rob Flickenger & Roger Weeks



HACK

#71

Wall Off Your Wireless

Build a tiny wireless firewall using a wide range of PC hardware and a BSD firewall.

If you've decided that small form factor PCs are for you, and have invested in such a [motherboard \[Hack #53\]](#) with the intentions of making your own access point, you're in the right hack. You certainly have other options for an operating system. [Pebble \[Hack #70\]](#) is a good example of a small Linux distribution designed for wireless applications. However, it has a couple negative factors: it hasn't been updated in over a year and configuration requires a pretty good knowledge of Linux daemons and how they like to be configured (with text files).

m0n0wall is an extremely tiny distribution of FreeBSD 4 that initially was designed as a tiny firewall, capable of running in a very small footprint. It has maintained those capabilities, weighing in at a surprisingly tiny 4.43MB for the image that runs on the [Soekris net4521 hardware \[Hack #53\]](#). In addition, the entire OS is configurable from a web browser. On the server, this is accomplished with a web server and PHP, and the entire configuration is stored in XML format.

Installation

You can download the software from <http://www.m0n0.ch/wall> in a number of formats. Specific images for Soekris hardware are available, as well as a generic PC image intended to run from hard disk or compact flash, and a bootable CDROM image. Specific instructions to get these latter images working can be found on the web site.

To get the Soekris hardware up and functioning with m0n0wall, you'll need a Compact Flash (CF) card at least 8MB in size, and some method of writing to that CF card from another working computer. There are a few ways to do this. You can use a [CF to IDE adapter \[Hack #54\]](#), a PC Card adapter that holds the CF card, or a USB CF reader.

You'll need to get your card reader or adapter hooked up to a Linux, BSD, or Windows box. Download the correct image for your hardware from the m0n0wall web site onto your chosen machine. Depending on which OS you choose, the methods for writing will be different.

BSD and Linux users will both use a *gzip* utility to decompress the image during the writing process. However, each OS has a different method of naming inserted rewritable media, so you will need to run *dmesg* from the command line to determine exactly the name of the device assigned to your CF card.

To write the image, BSD users should execute the following command:

```
gzcat net45xx-xxx.img | dd of=/dev/rad[n] bs=16k
```

Your CF card will be assigned a device name of something like */dev/rad3*. Make sure the command you execute has exactly the name of the device as determined from the output of *dmesg*.

Linux users will need to execute this command:

```
gunzip -c net45xx-xxx.img | dd of=/dev/hdX bs=16k
```

Again, you'll need to use the output of *dmesg* to determine what device name to use. If you have a USB CF reader, chances are good that your device will be loaded using a SCSI emulation device such as */dev/sda0*.

Users of both operating systems should ignore the error regarding "trailing garbage." This occurs because the software image is written with a digital signature.

Windows users need a special program that will write images directly to the media without the Windows disk system getting in the way. *Physdiskwrite* is just such a program, available from the m0n0wall web site at <http://www.m0n0.ch/wall/physdiskwrite.php>.

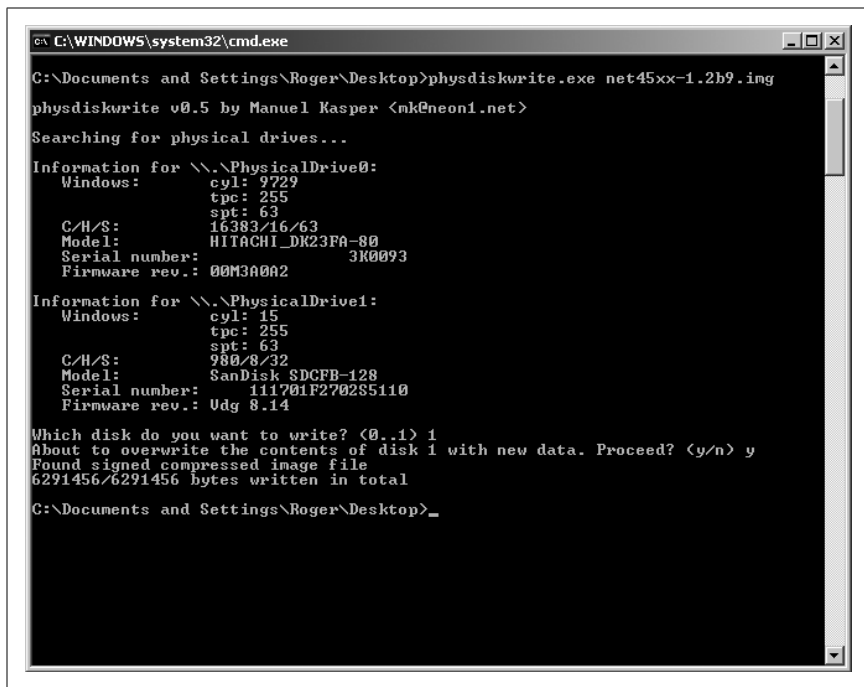
After inserting the CF on your Windows system, you can either call *Physdiskwrite* from a command window or drag the icon of the image file over the executable program icon. [Figure 5-11](#) shows a screenshot of the completed process. Note specifically that you will have to choose the correct physical drive! Choosing badly will overwrite another mounted physical disk on your system, and this could be very bad for you and your Windows install.

Congratulations! You've got m0n0wall written to your boot media. Unplug the CF card from your host system, insert it into the Soekris, and power up your new firewall with all of the network interfaces unplugged.

Configuration

Practically all of the m0n0wall configuration can be done from the web interface, but before that's possible you need to get the network interfaces configured. Since the Soekris is a headless (no display) device, you will need to connect to the serial port with a terminal program and view the console output.

If you're planning on spending any amount of time with the Soekris hardware, you'll want to invest either time or money in a null modem cable or adapter. This is not an ordinary serial cable, but one that has the Receive Data (RD) and Send Data (SD) pins crossed, as well as the Request To Send (RTS) and Clear To Send (CTS) pins. These cables can be purchased from



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Roger\Desktop>physdiskwrite.exe net45xx-1.2b9.ing
physdiskwrite v0.5 by Manuel Kasper <mk@neon1.net>
Searching for physical drives...

Information for \\.\PhysicalDrive0:
Windows:      cyl: 9729
              tpc: 255
              spt: 63
C/H/S:       16383/16/63
Model:       HITACHI_DK23FA-80
Serial number:      3K0093
Firmware rev.: 00M3A0A2

Information for \\.\PhysicalDrive1:
Windows:      cyl: 15
              tpc: 255
              spt: 63
C/H/S:       980/8/32
Model:       SanDisk_SDCFB-128
Serial number: 111701F2702S5110
Firmware rev.: Udg 8.14

Which disk do you want to write? <0..1> 1
About to overwrite the contents of disk 1 with new data. Proceed? <y/n> y
Found signed compressed image file
6291456/6291456 bytes written in total

C:\Documents and Settings\Roger\Desktop>_
```

Figure 5-11. Output from *Physdiskwrite*

just about any place that sells serial cables. If you want to make your own, check out the complete pinout reference at <http://www.nullmodem.com/NullModem.htm>.

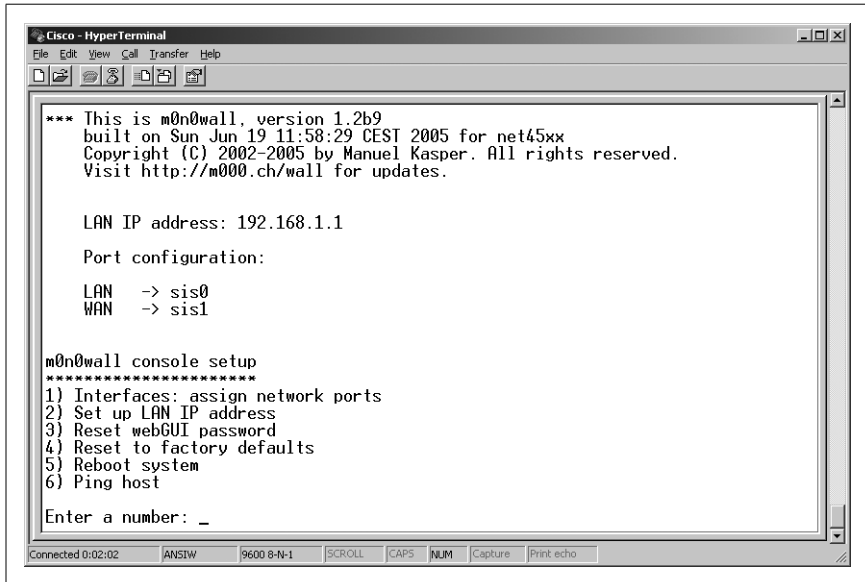
To connect to the console, Windows users can use the long-lived Hyperterminal program, found in Accessories → Communications. Linux and BSD users should investigate *minicom* or *ckermi*t. Whatever your terminal program, set your communication parameters for 19200,8,N,1.

The reason for this slightly unusual configuration is that, by default, the Soekris motherboards come set with a default console speed of 19200 bps. *m0n0wall* will function only with a 9600 bps speed, so you will have to change the Soekris defaults. Immediately after power-up, the Soekris will display a status screen. Press Ctrl-P to enter the ROM monitor mode. Enter the following commands:

```
set conspeed=9600
reboot
```

You'll now need to disconnect your terminal session, change the terminal speed to 9600 bps, and reconnect.

If you're successful, FreeBSD will boot, and then m0n0wall will present you with a menu, as shown in Figure 5-12.

The image shows a screenshot of a Cisco HyperTerminal window. The window title is "Cisco - HyperTerminal". The main content area displays the following text:

```
*** This is m0n0wall, version 1.2b9
built on Sun Jun 19 11:58:29 CEST 2005 for net45xx
Copyright (C) 2002-2005 by Manuel Kasper. All rights reserved.
Visit http://m000.ch/wall for updates.

LAN IP address: 192.168.1.1

Port configuration:

LAN   -> sis0
WAN   -> sis1

m0n0wall console setup
*****
1) Interfaces: assign network ports
2) Set up LAN IP address
3) Reset webGUI password
4) Reset to factory defaults
5) Reboot system
6) Ping host

Enter a number: _
```

The status bar at the bottom of the window shows "Connected 0:02:02", "ANSIW", "9600 8-N-1", "SCROLL", "CAPS", "NUM", "Capture", and "Print echo".

Figure 5-12. m0n0wall console configuration

The first three options in this menu are important:

1) *Interfaces: assign network ports*

Allows you to determine which interface on your Soekris will be assigned to LAN, WAN, and OPT ports of the firewall. OPT is a special port also referred to in other firewalls as a DMZ.

2) *Set up LAN IP address*

Lets you change the default LAN IP address from 192.168.1.1 to some other address of your choosing.

3) *Reset webGUI password*

Resets the web interface password if for some reason you have forgotten it.

Once you have your network interfaces configured the way you want them, m0n0wall will want to reboot the system. When the reboot is complete, you will be able to access the firewall using the LAN IP address. m0n0wall runs a DHCP server by default on the LAN, so you should be able to set your client computer for DHCP, obtain a lease, and point a web browser at <http://192.168.1.1>. If you changed the LAN IP address in the previous setup, alter this URL accordingly.

Using m0n0wall

For having such a small footprint, this operating system is positively packed with features. The webGUI configuration, shown in Figure 5-13, breaks these features down into sections, easily accessible from the navigation on the left side of the browser. When you first connect to the webGUI, you'll be prompted for a username (*admin*) and password (*mono*).



Figure 5-13. WebGUI configuration

First, click on General Setup and change the admin password. If your system is going to see a lot of public traffic, it's not a bad idea to alter the admin username to something else as well. Another security feature worth turning on is the HTTPS protocol for the webGUI, so all management traffic is encrypted. You can even alter the HTTPS port to a nonstandard one if you are really paranoid. (You *are* really paranoid about security, aren't you?) Lastly, you should set a hostname, your DNS servers, and an NTP server if you want to keep accurate time on your firewall.

Next you'll want to set up the WAN interface. If you've set up any wireless router before, this configuration will look familiar. m0n0wall supports DHCP, Static IP, PPPoE, PPTP, and something specific to Telstra BigPond cable Internet users. If you don't know what kind of WAN connection you're going to be using, now is a really good time to figure that out, because otherwise you have a pretty useless firewall.

At this point, you have a perfectly functional NAT firewall that assigns you a private IP address on the LAN segment, and uses the BSD *pf* firewall to do all of the packet filtering. But this is just scratching the surface!

There is a complete traffic-shaping section to m0n0wall. Choose Firewall → Traffic Shaper to define your own traffic shaping rules, but to get started quickly, click on the tab for Magic Shaper Wizard. This will let you set your downstream and upstream speeds for shaping. You can also set P2P traffic to the lowest priority, and share bandwidth evenly across all LAN users. After you apply the wizard's changes, you can go back and look at the other traffic shaper settings to get an idea of how to configure your own advanced shaping.

By default, m0n0wall sets itself up as a DNS forwarder and advertises its LAN address as the DNS server for all DHCP clients. This reduces DNS traffic on your WAN. You also have the option of enabling dynamic DNS through one of the online services that offer it, or setting up m0n0wall to talk to a RFC2136 compliant DNS server like BIND. You can configure all of these options from the Dynamic DNS menu.

If you've already looked at captive portals such as [NoCatAuth \[Hack #74\]](#) or [WiFiDog](#), you'll be pleased to find that m0n0wall includes its own configurable captive portal, listed under the Services menu.

There are too many captive portal features to cover here, but some of the most important ones include:

- Setting idle and hard timeout in minutes for clients
- Enabling authentication for clients via local users or RADIUS
- HTTPS logins for clients
- Custom HTML portal and error pages

m0n0wall also supports several kinds of VPN connections. Under the VPN menu, you can enable support for IPSec, PPTP, and OpenVPN.

The system Status menu has a number of pages you can use to monitor your system performance. The Interfaces page gives you In/Out packets and errors, the Traffic graph shows you utilization of your interfaces if you have

Wall Off Your Wireless

the Adobe SVG plug-in installed, and supported wireless cards using the *wi* driver show detailed connection information in the Wireless page.

If you want even more detailed information on system operation, the Diagnostics menu lets you view system logs, DHCP leases, IPSec connection details, as well as lets you backup and restore the system, reset to factory defaults, and reboot the system.

For the size and the price, you can't really beat m0n0wall for features on a small form factor PC.