A DHCP server for the local network

Relaxed Networking

Adding machines to a network means more configuration work. A centralized solution that uses a DHCP server to assign IP addresses and other values to the clients removes the administrative headaches.

BY MICHAEL RENNER

No matter whether you work for a business, are organizing a LAN party, setting up a network in shared accommodation or on your own multimedia home network, individually configuring each machine means a lot of hard work. Of course, a central system administrator can assign **IP addresses**, the **netmask**, the **default gateway**, and the **name server** this way. Try letting a group of people with varying levels of skill do this, and you are in for a big surprise.

Even experienced network administrators will experience a number of problems with this approach as the number of machines continues to rise. Each IP can be assigned only once to avoid collisions on the network. Even assuming that you have a carefully maintained list of the IPs assigned so far (and that you know where the list is), if you want to restructure your network (add a name server, use a different netmask or even assign a different IP to the router for access to public networks), this will involve manually re-configuring every single machine.

Guests with laptops need to know how and where to perform the necessary changes. Be honest: Do you really know how to set up the network configuration under MacOS 7.6? And do you want to admit to knowing how to set up the network on Windows 98?

The Dynamic Host Configuration Protocol (DHCP) provides the answer. To use DHCP you need to set up one

Listing 1: A simple, but still complete *dhcpd.conf*

```
01 default-lease-time 3600;
02 max-lease-time 14400;
03
04 subnet 192.168.2.0 netmask 255.255.255.0 {
05         range 192.168.2.50 192.168.2.69;
06         option domain-name-servers 194.25.2.129;
07         option broadcast-address 192.168.2.255;
08         option routers 192.168.2.1;
09 }
```

GLOSSARY

IP address: *Each machine on the network is identified by means of a unique 32 bit IP address. To improve readability, addresses use dotted decimal notation, where each segment of the address can be a number between 0 and 255. In addition to official IP addresses that ensure global accessibility, a number of address blocks have been assigned for use on private networks. These addresses are not accessible via public networks.*

Netmask: *The netmask is used to define the boundaries of a network segment. Machines use bit masks to ascertain the IP addresses on their own subnets. Box 1 shows an example.*

Name server: *A Domain Name Server provides facilities for translating symbolic host names (for example linux-magazine.com) to IP addresses for example 62.245.157.219) and vice-versa. DNS is a distributed, hierarchical database system that does not require each name server to know each hostname, but instead allows a machine to query other machines.*

Default gateway: *A router on the local network to which a machine will send any packets not destined for local addresses.*

machine as a DHCP server. After doing so, this machine will be responsible for managing the network configurations of the other machines on the network. Besides traditional parameters, you can allow the DHCP server to define values specific to the local network, for example, a **timeserver**, or even the name of a **WINS** server.

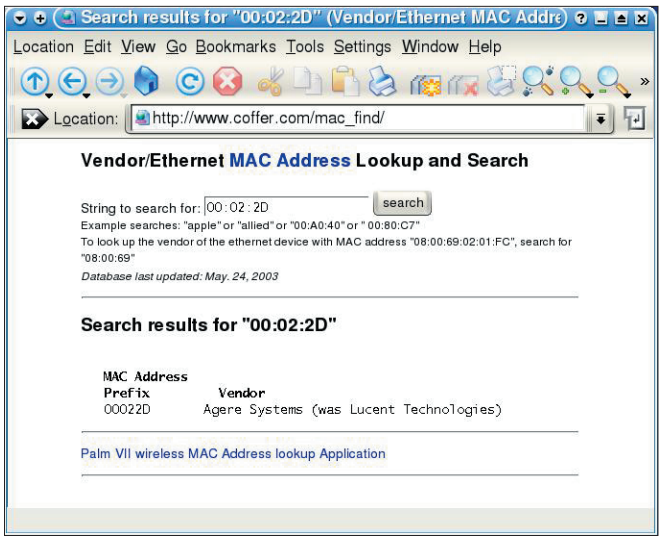

Figure 1: You can use MAC find to map MAC addresses to card manufacturers.

Many hosts ...

What does this look like in a practical application? Let's look at a small private network. In this case, "private" means that the network uses a pool of non-routable IP addresses. To avoid confusion, these addresses will work on local networks (LANs), but not on the public Internet.

Really big private networks (so-called Class A networks) can use the address range 10.0.0.0 through 10.255.255.255 for private networks. A medium sized, private Class B network can use the address range 172.16.0.0 through 172.31.255.255. If your network is of a more modest size, you can use the 65023 addresses in the range 192.168.0.0 through 192.168.255.255.

Small businesses and home networks will be perfectly happy with a Class C network. I'm going to use my house number in this example and have assigned the IP 192.168.2.0 to the network (this is useful if you intend to set up a network with your neighbors).

... and just one server

The machine that will be assigning all these IPs needs a DHCP server program. There are not too many free implementations, so let's opt for the ISC DHCP server, *dhcpd*, like most people do.

If you really want to build the program yourself, the source code is available from [1]. Most people will be happy enough to install the package for their distro. Debian users can enter the following:

```
apt-get install dhcp
```

Users with other distributions will typically find a suitable RPM package on their distribution CDs or DVDs. If not, try *http://rpmfind.net/*; search for *DHCP*.

Following the install, you should have a *dhcpd.conf* configuration file in your */etc/* directory. The file is well-documented. This should allow you to modify the values in the following examples to reflect your own environment. Take a while to think about the number of computers you need to support simultaneously on your network. If you are organizing a LAN party, this may be a large number. A home network typically has less than ten machines.

The example in Listing 1 assumes that no more than 20 machines need simultaneous access to the network. We want *dhcpd* to assign IP addresses from the range *192.168.2.50* through *192.168.2.69*. Also, we will be using the *domain-name-servers* option to define a name server, and *routers* to specify the Internet gateway. If you do not have a name

Table 1: dhcpd Settings

Entry	Parameter	Meaning
default-lease-time	Time in seconds	Specifies the validity period for the assigned values. The client needs to re-request the IP address within this period. If the client fails to do so, the IP address can be assigned to another machine.
max-lease-time	Time in seconds	Specifies the maximum validity period of the assigned values. If the client requests a particularly long *default-lease-time*, this parameter sets the max value.
subnet	Network address	Network segment to which this configuration applies (see Box 1).
netmask	Networ mask	Mask for this network segment (see Box 1).
range	Lowest and highest IP address	The pool of addresses to be assigned by the DHCP server.
fixed-address	IP address or hostname	A permanent address to be assigned to a specific client.
filename	Filename	Boot image for a specific client (see section "Advanced DHCP Techniques").
hardware ethernet	MAC address	Hardware address of client.

GLOSSARY

Timeserver: *Provides the authoritative time on a network. The official time in the UK is measured from the Greenwich Meridian Line at the Royal Observatory in Greenwich. In the US, two time agencies, the National Institute of Standards and Technology (NIST) and its military counterpart, the US Naval Observatory (USNO), provide official timekeeping (http://www.time.gov). In addition to these sources, there are a number of more or less accurate timeservers on the Web.*

WINS: *Microsoft's "Windows Internet Naming Service" maps the NetBIOS name of a Windows computer to its IP address. A WINS server is a kind of special purpose name server. If you have a lot of Windows boxes on your network, the WINS server can prevent unwanted* **broadcasts**. *The Samba project provides WINS support.*

Broadcast: *Undirected broadcasts are used to reach multiple hosts without needing any information on these hosts. A broadcast can save bandwidth if you need to talk to a large number of machines, but typically, broadcasting will add to the network load and should be avoided.*

server on your LAN, use your Internet provider's name server instead.

Note the curly brackets that enclose the definition for a network segment. Each entry in the configuration is terminated by a semicolon.

Freely Configurable

Generally speaking, there are two types of entries in the configuration file: ones that start with the *option* keyword, and the rest. The options (Table 2 provides an overview) are passed to the requesting machines (referred to as clients) as is by the *dhcpd*. Whether or not the client can actually use this information depends on the client operating system. Admins might want to define scripts to specify how this information will be processed.

The remaining entries are used by the server process, *dhcpd*, itself – Table 1 again gives an overview.

Permanent Assignments

If you decide to use DHCP throughout your network, there will typically be a few computers that need a static IP. After all, it does not make sense to have to guess the current address of the gateway or multimedia server before you can access it.

The */etc/dhcpd.conf* file is used for client-dependent address assignments. *dhcpd* evaluates the unique MAC address to be able to identify the requesting machine and assign the correct address.

The MAC address is printed on PCMCIA network adapters or Wireless LAN USB sticks. If you cannot read the label on the NIC, you can query the MAC address. Unix-style operating systems use the *ifconfig* command to do this, Microsoft operating systems have a command line tool called *ipconfig*. Listing 2 shows an example of a Linux machine whose MAC or "hardware" address is *00:02:2D:34:90:85*.

Don't worry if you cannot access the machine directly – asssuming that the DHCP server has assigned this machine an arbitrary address. *dhcpd* keeps a log of address assignments and lease periods in */var/lib/dhcp/dhcpd.leases*.

Another approach would be to *ping* all the hosts on the network to provoke a reaction from them. After having done so, you can enter *arp -a* to find out the hosts and MAC addresses known to your localhost (see Listing 3). You need to *ping* first, as the *arp* table, which contains the MAC addresses of all known hosts on the local network, will only include details for machines that your computer has recently talked to.

Which MAC?

However, you still need to guess which of these MAC addresses is the one you are looking for. In some cases it might help to refer to a special database [2] that reveals the MAC addresses assigned to network card manufacturers (see Fig-

Box 1: IP address, netmask and network address

Besides the IP address, the configuration of a network device includes the broadcast address and netmask. The netmask is used to subdivide a network into smaller subnets. To understand the relationship between IP addresses and the network mask, we need to inspect the bits at the lowest level.

To do so, let's take each of the decimal numbers separated by the dots in the address and convert it to its binary equivalent:

```
255=1*2^7+1*2^6+1*2^5+1*2^4+1*2^3
+1*2^2+1*2^1+1*2^0
```

Thus, a netmask of 255.255.255.0 in dotted decimal format becomes the binary netmask:

```
11111111.11111111.11111111.000000
00
```

Of course, this has not involved a great deal of math, as we have simply converted a number in one base to another. The 24 ones are interesting – the network is referred to as a /24 (slash 24) network.

Applying the same technique, we can see that a dotted decimal IP of 192.168.2.3 will resolve to binary

```
11000000.10101000.00000010.000000
11
```

The network address is a bitwise AND of the IP address and the netmask. This operation leaves a 1 where the IP and the netmask both have a 1:

```
11000000.10101000.00000010.000000
00
```

The network address is thus 192.168.2.0 in dotted decimal.

As the zero IP address is reserved for the network itself, and a further address (typically .255) is reserved for broadcasts, the network in our example can accommodate 254 machines.

Listing 2: Using *ifconfig* to discover a MAC address

```
01 renner@lyra:~$ /sbin/ifconfig eth0
02 eth0      Link encap:Ethernet  HWaddr 00:02:2D:34:90:85
03           inet addr:10.32.130.79  Bcast:10.32.135.255
   Mask:255.255.248.0
04           UP BROADCAST NOTRAILERS RUNNING MULTICAST  MTU:1500
   Metric:1
05           RX packets:15695 errors:0 dropped:0 overruns:0 frame:0
06           TX packets:10988 errors:204 dropped:0 overruns:0 carrier:0
07           collisions:0 txqueuelen:100
08           RX bytes:5201433 (4.9 MiB)  TX bytes:1559490 (1.4 MiB)
09           Interrupt:10 Base address:0x100
```

Table 2: Client values

Entry (without *option*)	Parameter	Meaning
routers	Hostname or IP address	Routers or gateways for Internet access.
domain-name-servers	Hostname or IP address	Domain name server.
host-name	Hostname	Hostname of client.
ntp-servers	Hostname or IP address	Timeserver for time synchronization.
netbios-node-type	1, 2, 4, or 8 (recommended)	Name resolution approach for Windows. *1* means broadcast, *2* **unicast**, *4* mixed mode (broadcast first, then try unicast), and *8* is a hybrid mode, that unicasts first, before attempting to broadcast.
netbios-name-servers	Hostname	WINS server for Internet naming service resolution on Windows.
domain-name	Domain name	Name of network domain.
nis-domain	Domain name	Name of **NIS** domain.
nis-servers	Hostname or IP address	NIS server.
subnet-mask	Netmask	Netmask of the network segment.

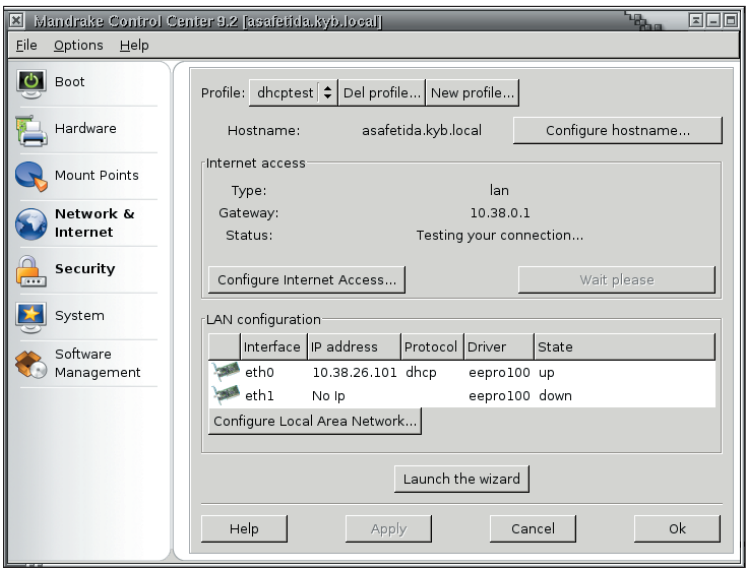

Figure 2: Enabling the DHCP client on Mandrake.

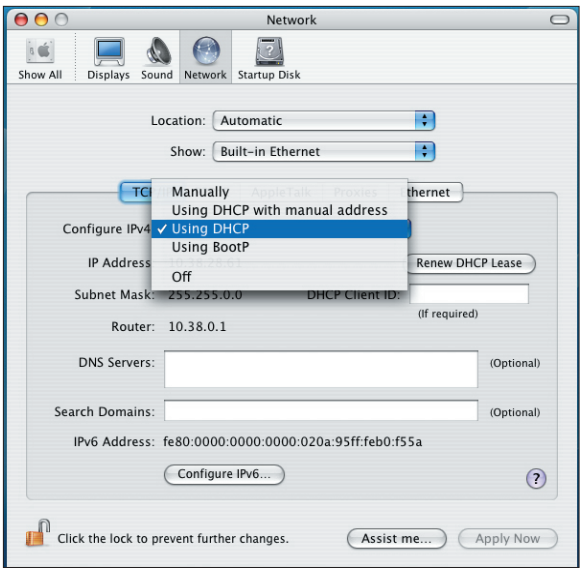

Figure 3: MacOS X supports DHCP.

ure 1). This knowledge could speed up the identification process.

After discovering the hardware address, you can add a static IP mapping to your *dhcpd* configuration:

```
host lyraA  {
hardware ethernet ➥
00:02:2D:34:90:85;
fixed-address lyra.mtr.mynet;
}
```

If the computer has multiple NICs (e.g. an additional wireless LAN card), you can assign the same hostname to both MAC addresses:

```
host lyraB  {
hardware ethernet ➥
00:80:C7:C1:3D:76;
fixed-address lyra.mtr.mynet;
}
```

In some cases, you can even swap the NIC while a machine is running without interrupting an existing connection! Instead of a hostname, which requires working name resolution facilities, you can assign an IP address.

Advanced DHCP Techniques

DHCP can do more! DHCP tells a diskless client that downloads its operating system off the network the name of its **boot image**. This is achieved by using the Trivial File Transfer Protocol, TFTP, a subset of FTP, to transfer the information across the network. Listing 4 shows an example where a SGI Indy workstation is assigned its Linux kernel via DHCP. This kind of boot procedure not only occurs in large clusters, which are used to solve mathematical problems, for example, but also in computer pools at universities or colleges.

Listing 3: Discovering MAC addresses with *ping* and *arp*

```
01 renner@lyra:~$ ping -c3 192.168.2.0
02 PING 192.168.2.0 (192.168.2.0): 56 data bytes
03 64 bytes from 192.168.2.1: icmp_seq=0 ttl=64 time=0.2 ms
04 64 bytes from 192.168.2.52: icmp_seq=0 ttl=64 time=3.5 ms (DUP!)
05 64 bytes from 192.168.2.53: icmp_seq=0 ttl=64 time=4.2 ms (DUP!)
06 64 bytes from 192.168.2.62: icmp_seq=0 ttl=64 time=4.7 ms (DUP!)
07 [...]
08 renner@lyra:~$ /usr/sbin/arp -a
09 ? (192.168.2.1) at 00:03:E3:00:18:F1 [ether] on eth0
10 ? (192.168.2.52) at 00:30:05:55:02:ED [ether] on eth0
11 ? (192.168.2.53) at 00:0C:6E:1F:32:C4 [ether] on eth0
12 ? (192.168.2.62) at 00:30:05:55:03:7F [ether] on eth0
```

GLOSSARY

Unicast: *Each client is sent a copy of a file by the server. Point to point connections of this type are easy to configure, but they do place a heavy load on the server if there are a lot of clients on the network.*

NIS: *The "Network Information Service" facilitates the propagation of configurations on a network. A central NIS server provides information on login and password credentials, home directories, group assignments, and hostnames. This allows the NIS server to complete the entries in arbitrary, client-side configuration files, e.g. /etc/passwd, /etc/groups, or /etc/hosts. A file called /etc/nsswitch.conf specifies whether the NIS server supplies information for configuration files, and if so, what kind of information. The alternative approach is to have the DHCP server provide details of the NIS server on the local network.*

MAC address: *The Media Access Control, or more simply, the Ethernet address, is a six byte hexadecimal number. The MAC address is hard-coded into each Ethernet device, and allows the device to be identified uniquely. Each manufacturer has a contingent of MAC addresses and assigns each address only once.*

Boot image: *Specially adapted Unix or Linux kernel which is downloaded off the network and launched. This is mainly seen in environments with diskless clients, which mount their root file system (/) across the network. Network-based boot images allow admins to install large numbers of computers without resorting to local media (CDs, floppies).*

Listing 4: Assigning a boot image

```
01 host indy {
02    filename "indy_r4k_tftpboot.img";
03    hardware ethernet 08:00:69:08:58:40;
04    fixed-address 192.168.2.12;
05    server-name "cassiopeia.mtr.mynet";
06    option host-name "indy";
07    option domain-name "mtr.mynet";
08    option domain-name-servers 192.168.2.53;
09    option routers 192.168.2.1;
10 }
```

This can only work if the *tftpd* server is enabled in */etc/inetd.conf*. To allow the server to find the specified boot image (*indy_r4k_tftpboot.img* in our example), you need to pass the directory with the files (such as */boot*) to the server:

```
tftp dgram udp wait nobody ➥
/usr/sbin/tcpd /usr/sbin➥
/in.tftpd /boot
```

Distributions that use the more modern *xinetd*, rather than the *inetd* superserver, should have an */etc/xinetd.d/tftp* file like the one shown in Listing 5.

A single DHCP server can support multiple network segments at the same time. To allow this, you need to create multiple *subnet* sections in your configuration. Options enclosed in curly brackets, such as details on the name server or the NIS domain, must be unique. In practical applications each network segment is typically assigned to one of the DHCP server's NICs.

dhcpd has to be restarted to recognize the configuration changes. As changes of this kind only happen very occasionally, admins can normally sit back and relax when new machines are connected to the network.

Client-Side

The new arrivals do not need a lot of configuration work. Instead of entering a static IP, you simply enable DHCP. Depending on the operating system (and possibly the distribution), there are various approaches to doing so. If you have Debian, you can add a line like the following

```
iface eth0 inet dhcp
```

to the */etc/network/interfaces* file. Other distributions typically have a GUI-based configuration tool. Mandrake uses the wizard shown in Figure 2.

Windows users need to check the *Network* tool in the *Control Panel*. Users with classic MacOS systems need to access *Control Panel – TCP/IP* in the Apple menu. MacOS X (see Figure 3) enables DHCP in *System Preferences / Internet & Network*.

Finally

There is not a big selection of DHCP client programs for Linux. One of them is the *pump* client, which is particularly popular with mini-distros on account of its tiny footprint. The larger footprint alternatives, such as *dhcp-client* and *dhcpcd* have more functions, such as the ability to launch a script after completing IP configuration.

There are a few pitfalls with older *dhcp* packages on the kernel 2.6. This combination will work, but */sbin/dhclient* script checks the kernel version for *dhcp-client*. As it does not recognize the kernel 2.6, it quits at this point. As DHCP only needs to distinguish between kernel 2.0 or later versions, you can easily modify the script to recognize 2.6 as a valid kernel version:

```
2.[123456].*)
exec /sbin/dhclient➥
-2.2.x -q "$@"
;;
```

Simply add a *6* in the right place and you should be soon up and running without any further problems. ■

Listing 5: File /etc/xinetd.d/tftp

```
01 # default: off
02
03 service tftp
04 {
05 disable = no
06 socket_type = dgram
07 protocol = udp
08 wait = yes
09 user = root
10 server = /usr/sbin/in.tftpd
11 server_args = -s /boot
12 }
```

INFO

[1] DHCP Server: *http://www.isc.org/index.pl?/sw/dhcp/*

[2] MAC database: *http://www.coffer.com/mac_find/*