

Internet Systems Consortium DHCP Distribution  
Version 4.2.3  
18 October 2011

README FILE

You should read this file carefully before trying to install or use the ISC DHCP Distribution.

TABLE OF CONTENTS

1	WHERE TO FIND DOCUMENTATION
2	RELEASE STATUS
3	BUILDING THE DHCP DISTRIBUTION
3.1	UNPACKING IT
3.2	CONFIGURING IT
3.2.1	DYNAMIC DNS UPDATES
3.2.2	LOCALLY DEFINED OPTIONS
3.3	BUILDING IT
4	INSTALLING THE DHCP DISTRIBUTION
5	USING THE DHCP DISTRIBUTION
5.1	FIREWALL RULES
5.2	LINUX
5.2.1	IF_TR.H NOT FOUND
5.2.2	SO_ATTACH_FILTER UNDECLARED
5.2.3	PROTOCOL NOT CONFIGURED
5.2.4	BROADCAST
5.2.6	IP BOOTP AGENT
5.2.7	MULTIPLE INTERFACES
5.3	SCO
5.4	HP-UX
5.5	ULTRIX
5.6	FreeBSD
5.7	NeXTSTEP
5.8	SOLARIS
5.8.1	Solaris 11
5.8.2	Other Solaris Items
5.9	AIX
5.10	MacOS X
6	SUPPORT
6.1	HOW TO REPORT BUGS

WHERE TO FIND DOCUMENTATION

Documentation for this software includes this README file, the RELNOTES file, and the manual pages, which are in the server, common, client and relay subdirectories. The README file (this file) includes late-breaking operational and system-specific information that you should read even if you don't want to read the manual pages, and that you should **\*certainly\*** read if you run into trouble. Internet standards relating to the DHCP protocol are listed in the References document that is available in html, txt and xml formats in doc/ subdirectory. You will have the best luck reading the manual pages if you build this software and then install it, although you can read them directly out of the distribution if you need to.

DHCP server documentation is in the dhcpd man page. Information about

the DHCP server lease database is in the dhcpd.leases man page. Server configuration documentation is in the dhcpd.conf man page as well as the dhcp-options man page. A sample DHCP server configuration is in the file server/dhcpd.conf. The source for the dhcpd, dhcpd.leases and dhcpd.conf man pages is in the server/ subdirectory in the distribution. The source for the dhcp-options.5 man page is in the common/ subdirectory.

DHCP Client documentation is in the dhclient man page. DHCP client configuration documentation is in the dhclient.conf man page and the dhcp-options man page. The DHCP client configuration script is documented in the dhclient-script man page. The format of the DHCP client lease database is documented in the dhclient.leases man page. The source for all these man pages is in the client/ subdirectory in the distribution. In addition, the dhcp-options man page should be referred to for information about DHCP options.

DHCP relay agent documentation is in the dhcrelay man page, the source for which is distributed in the relay/ subdirectory.

To read installed manual pages, use the man command. Type "man page" where page is the name of the manual page. This will only work if you have installed the ISC DHCP distribution using the ``make install'' command (described later).

If you want to read manual pages that aren't installed, you can type ``nroff -man page |more'' where page is the filename of the unformatted manual page. The filename of an unformatted manual page is the name of the manual page, followed by '.', followed by some number - 5 for documentation about files, and 8 for documentation about programs. For example, to read the dhcp-options man page, you would type ``nroff -man common/dhcp-options.5 |more'', assuming your current working directory is the top level directory of the ISC DHCP Distribution.

Please note that the pathnames of files to which our manpages refer will not be correct for your operating system until after you iterate 'make install' (so if you're reading a manpage out of the source directory, it may not have up-to-date information).

#### RELEASE STATUS

This is ISC DHCP 4.2.3, a maintenance release containing patches.

In this release, the DHCPv6 server should be fully functional on Linux, Solaris, or any BSD. The DHCPv6 client should be similarly functional except on Solaris.

The DHCPv4 server, relay, and client, should be fully functional on Linux, Solaris, any BSD, HPUX, SCO, NextSTEP, and Irix.

If you are running the DHCP distribution on a machine which is a firewall, or if there is a firewall between your DHCP server(s) and DHCP clients, please read the section on firewalls which appears later in this document.

If you wish to run the DHCP Distribution on Linux, please see the

Linux-specific notes later in this document. If you wish to run on an SCO release, please see the SCO-specific notes later in this document. You particularly need to read these notes if you intend to support Windows 95 clients. If you are running HP-UX or Ultrix, please read the notes for those operating systems below. If you are running NeXTSTEP, please see the notes on NeXTSTEP below.

If you start dhcpd and get a message, "no free bpf", that means you need to configure the Berkeley Packet Filter into your operating system kernel. On NetBSD, FreeBSD and BSD/os, type ``man bpf'' for information. On Digital Unix, type ``man pfil''.

## BUILDING THE DHCP DISTRIBUTION

### UNPACKING IT

To build the DHCP Distribution, unpack the compressed tar file using the tar utility and the gzip command - type something like:

```
gunzip dhcp-4.2.3.tar.gz
tar xvf dhcp-4.2.3.tar
```

### CONFIGURING IT

Now, cd to the dhcp-4.2.3 subdirectory that you've just created and configure the source tree by typing:

```
./configure
```

If the configure utility can figure out what sort of system you're running on, it will create a custom Makefile for you for that system; otherwise, it will complain. If it can't figure out what system you are using, that system is not supported - you are on your own.

### DYNAMIC DNS UPDATES

A fully-featured implementation of dynamic DNS updates is included in this release. It uses libraries from BIND and, to avoid issues with different versions, includes the necessary BIND version. The appropriate BIND libraries will be compiled and installed in the bind subdirectory as part of the make step. In order to build the necessary libraries you will need to have "gmake" available on your build system.

There is documentation for the DDNS support in the dhcpd.conf manual page - see the beginning of this document for information on finding manual pages.

### LOCALLY DEFINED OPTIONS

In previous versions of the DHCP server there was a mechanism whereby options that were not known by the server could be configured using a name made up of the option code number and an identifier: "option-*nnn*". This is no longer supported, because it is not future-proof. Instead, if you want to use an option that the server doesn't

know about, you must explicitly define it using the method described in the dhcp-options man page under the DEFINING NEW OPTIONS heading.

#### BUILDING IT

Once you've run configure, just type ``make'', and after a while you should have a dhcp server. If you get compile errors on one of the supported systems mentioned earlier, please let us know. If you get warnings, it's not likely to be a problem - the DHCP server compiles completely warning-free on as many architectures as we can manage, but there are a few for which this is difficult. If you get errors on a system not mentioned above, you will need to do some programming or debugging on your own to get the DHCP Distribution working.

#### INSTALLING THE DHCP DISTRIBUTION

Once you have successfully gotten the DHCP Distribution to build, you can install it by typing ``make install''. If you already have an old version of the DHCP Distribution installed, you may want to save it before typing ``make install''.

#### USING THE DHCP DISTRIBUTION

##### FIREWALL RULES

If you are running the DHCP server or client on a computer that's also acting as a firewall, you must be sure to allow DHCP packets through the firewall. In particular, your firewall rules must allow packets from IP address 0.0.0.0 to IP address 255.255.255.255 from UDP port 68 to UDP port 67 through. They must also allow packets from your local firewall's IP address and UDP port 67 through to any address your DHCP server might serve on UDP port 68. Finally, packets from relay agents on port 67 to the DHCP server on port 67, and vice versa, must be permitted.

We have noticed that on some systems where we are using a packet filter, if you set up a firewall that blocks UDP port 67 and 68 entirely, packets sent through the packet filter will not be blocked. However, unicast packets will be blocked. This can result in strange behaviour, particularly on DHCP clients, where the initial packet exchange is broadcast, but renewals are unicast - the client will appear to be unable to renew until it starts broadcasting its renewals, and then suddenly it'll work. The fix is to fix the firewall rules as described above.

##### PARTIAL SERVERS

If you have a server that is connected to two networks, and you only want to provide DHCP service on one of those networks (e.g., you are using a cable modem and have set up a NAT router), if you don't write any subnet declaration for the network you aren't supporting, the DHCP server will ignore input on that network interface if it can. If it can't, it will refuse to run - some operating systems do not have the capability of supporting DHCP on machines with more than one interface, and ironically this is the case even if you don't want to provide DHCP service on one of those interfaces.

## LINUX

There are three big LINUX issues: the all-ones broadcast address, Linux 2.1 ip\_bootp\_agent enabling, and operations with more than one network interface. There are also two potential compilation/runtime problems for Linux 2.1/2.2: the "SO\_ATTACH\_FILTER undeclared" problem and the "protocol not configured" problem.

### LINUX: PROTOCOL NOT CONFIGURED

If you get the following message, it's because your kernel doesn't have the linux packetfilter or raw packet socket configured:

Make sure CONFIG\_PACKET (Packet socket) and CONFIG\_FILTER (Socket Filtering) are enabled in your kernel configuration

If this happens, you need to configure your Linux kernel to support Socket Filtering and the Packet socket, or to select a kernel provided by your Linux distribution that has these enabled (virtually all modern ones do by default).

### LINUX: BROADCAST

If you are running a recent version of Linux, this won't be a problem, but on older versions of Linux (kernel versions prior to 2.2), there is a potential problem with the broadcast address being sent incorrectly.

In order for dhcpd to work correctly with picky DHCP clients (e.g., Windows 95), it must be able to send packets with an IP destination address of 255.255.255.255. Unfortunately, Linux changes an IP destination of 255.255.255.255 into the local subnet broadcast address (here, that's 192.5.5.223).

This isn't generally a problem on Linux 2.2 and later kernels, since we completely bypass the Linux IP stack, but on old versions of Linux 2.1 and all versions of Linux prior to 2.1, it is a problem - pickier DHCP clients connected to the same network as the ISC DHCP server or ISC relay agent will not see messages from the DHCP server. It **\*is\*** possible to run into trouble with this on Linux 2.2 and later if you are running a version of the DHCP server that was compiled on a Linux 2.0 system, though.

It is possible to work around this problem on some versions of Linux by creating a host route from your network interface address to 255.255.255.255. The command you need to use to do this on Linux varies from version to version. The easiest version is:

```
route add -host 255.255.255.255 dev eth0
```

On some older Linux systems, you will get an error if you try to do this. On those systems, try adding the following entry to your /etc/hosts file:

```
255.255.255.255 all-ones
```

Then, try:

```
route add -host all-ones dev eth0
```

Another route that has worked for some users is:

```
route add -net 255.255.255.0 dev eth0
```

If you are not using eth0 as your network interface, you should specify the network interface you **\*are\*** using in your route command.

#### LINUX: IP BOOTP AGENT

Some versions of the Linux 2.1 kernel apparently prevent dhcpd from working unless you enable it by doing the following:

```
echo 1 >/proc/sys/net/ipv4/ip_bootp_agent
```

#### LINUX: MULTIPLE INTERFACES

Very old versions of the Linux kernel do not provide a networking API that allows dhcpd to operate correctly if the system has more than one broadcast network interface. However, Linux 2.0 kernels with version numbers greater than or equal to 2.0.31 add an API feature: the SO\_BINDTODEVICE socket option. If SO\_BINDTODEVICE is present, it is possible for dhcpd to operate on Linux with more than one network interface. In order to take advantage of this, you must be running a 2.0.31 or greater kernel, and you must have 2.0.31 or later system headers installed **\*before\*** you build the DHCP Distribution.

We have heard reports that you must still add routes to 255.255.255.255 in order for the all-ones broadcast to work, even on 2.0.31 kernels. In fact, you now need to add a route for each interface. Hopefully the Linux kernel gurus will get this straight eventually.

Linux 2.1 and later kernels do not use SO\_BINDTODEVICE or require the broadcast address hack, but do support multiple interfaces, using the Linux Packet Filter.

#### LINUX: OpenWrt

DHCP 4.1 has been tested on OpenWrt 7.09 and 8.09. In keeping with standard practice, client/scripts now includes a dhclient-script file for OpenWrt. However, this is not sufficient by itself to run dhcp on OpenWrt; a full OpenWrt package for DHCP is available at <ftp://ftp.isc.org/isc/dhcp/dhcp-4.1.0-openwrt.tar.gz>

#### LINUX: 802.1q VLAN INTERFACES

If you're using 802.1q vlan interfaces on Linux, it is necessary to vconfig the subinterface(s) to rewrite the 802.1q information out of packets received by the dhcpd daemon via LPF:

```
vconfig set_flag eth1.523 1 1
```

Note that this may affect the performance of your system, since the

Linux kernel must rewrite packets received via this interface. For more information, consult the vconfig man pages.

#### SCO

ISC DHCP will now work correctly on newer versions of SCO out of the box (tested on OpenServer 5.05b, assumed to work on UnixWare 7).

Older versions of SCO have the same problem as Linux (described earlier). The thing is, SCO **\*really\*** doesn't want to let you add a host route to the all-ones broadcast address.

You can try the following:

```
ifconfig net0 xxx.xxx.xxx.xxx netmask 0xNNNNNNNN broadcast 255.255.255.255
```

If this doesn't work, you can also try the following strange hack:

```
ifconfig net0 alias 10.1.1.1 netmask 8.0.0.0
```

Apparently this works because of an interaction between SCO's support for network classes and the weird netmask. The 10.\* network is just a dummy that can generally be assumed to be safe. Don't ask why this works. Just try it. If it works for you, great.

#### HP-UX

HP-UX has the same problem with the all-ones broadcast address that SCO and Linux have. One user reported that adding the following to /etc/rc.config.d/netconf helped (you may have to modify this to suit your local configuration):

```
INTERFACE_NAME[0]=lan0
IP_ADDRESS[0]=1.1.1.1
SUBNET_MASK[0]=255.255.255.0
BROADCAST_ADDRESS[0]="255.255.255.255"
LANCONFIG_ARGS[0]="ether"
DHCP_ENABLE[0]=0
```

#### ULTRIX

Now that we have Ultrix packet filter support, the DHCP Distribution on Ultrix should be pretty trouble-free. However, one thing you do need to be aware of is that it now requires that the pfilter device be configured into your kernel and present in /dev. If you type ``man packetfilter'', you will get some information on how to configure your kernel for the packet filter (if it isn't already) and how to make an entry for it in /dev.

#### FreeBSD

Versions of FreeBSD prior to 2.2 have a bug in BPF support in that the ethernet driver swaps the ethertype field in the ethernet header downstream from BPF, which corrupts the output packet. If you are running a version of FreeBSD prior to 2.2, and you find that dhcpd can't communicate with its clients, you should #define BROKEN\_FREEBSD\_BPF in site.h and recompile.

Modern versions of FreeBSD include the ISC DHCP 3.0 client as part of the base system, and the full distribution (for the DHCP server and relay agent) is available from the Ports Collection in /usr/ports/net/isc-dhcp3, or as a package on FreeBSD installation CDROMs.

#### NeXTSTEP

The NeXTSTEP support uses the NeXTSTEP Berkeley Packet Filter extension, which is not included in the base NextStep system. You must install this extension in order to get dhcpd or dhclient to work.

#### SOLARIS

There are two known issues seen when compiling using the Sun compiler.

The first is that older Sun compilers generate an error on some of our uses of the flexible array option. Newer versions only generate a warning, which can be safely ignored. If you run into this error ("type of struct member "buf" can not be derived from structure with flexible array member"), upgrade your tools to Oracle Solaris Studio (previously Sun Studio) 12 or something newer.

The second is the interaction between the configure script and the makefiles for the Bind libraries. Currently we don't pass all environment variables between the DHCP configure and the Bind configure.

If you attempt to specify the compiler you wish to use like this:

```
CC=/opt/SUNWspro/bin/cc ./configure
```

"make" may not build the Bind libraries with that compiler.

In order to use the same compiler for Bind and DHCP we suggest the following commands:

```
CC=/opt/SUNWspro/bin/cc ./configure
CC=/opt/SUNWspro/bin/cc make
```

#### Solaris 11

We have integrated a patch from Oracle to use sockets instead of DLPI on Solaris 11. This functionality was written for use with Solaris Studio 12.2 and requires the system/header package.

By default this code is disabled in order to minimize disruptions for current users. In order to enable this code you will need to enable both USE\_SOCKETS and USE\_V4\_PKTINFO as part of the configuration step. The command line would be something like:

```
./configure --enable-use-sockets --enable-ipv4-pktinfo
```

#### Other Solaris Items

One problem which has been observed and is not fixed in this patchlevel has to do with using DLPI on Solaris machines. The symptom

of this problem is that the DHCP server never receives any requests. This has been observed with Solaris 2.6 and Solaris 7 on Intel x86 systems, although it may occur with other systems as well. If you encounter this symptom, and you are running the DHCP server on a machine with a single broadcast network interface, you may wish to edit the includes/site.h file and uncomment the #define USE\_SOCKETS line. Then type ``make clean; make''. As an alternative workaround, it has been reported that running 'snoop' will cause the dhcp server to start receiving packets. So the practice reported to us is to run snoop at dhcpd startup time, with arguments to cause it to receive one packet and exit.

```
snoop -c 1 udp port 67 > /dev/null &
```

The DHCP client on Solaris will only work with DLPI. If you run it and it just keeps saying it's sending DHCPREQUEST packets, but never gets a response, you may be having DLPI trouble as described above. If so, we have no solution to offer at this time, aside from the above workaround which should also work here. Also, because Solaris requires you to "plumb" an interface before it can be detected by the DHCP client, you must either specify the name(s) of the interface(s) you want to configure on the command line, or must plumb the interfaces prior to invoking the DHCP client. This can be done with ``ifconfig iface plumb'', where iface is the name of the interface (e.g., ``ifconfig hme0 plumb'').

It should be noted that Solaris versions from 2.6 onward include a DHCP client that you can run with ``/sbin/ifconfig iface dhcp start'' rather than using the ISC DHCP client, including DHCPv6. Consequently, we don't believe there is a need for the client to run on Solaris, and have not engineered the needed DHCPv6 modifications for the dhclient-script. If you feel this is in error, or have a need, please contact us.

#### AIX

The AIX support uses the BSD socket API, which cannot differentiate on which network interface a broadcast packet was received; thus the DHCP server and relay will work only on a single interface. (They do work on multi-interface machines if configured to listen on only one of the interfaces.)

We have reports of Windows XP clients having difficulty retrieving addresses from a server running on an AIX machine. This issue was traced to the client requiring messages be sent to the all ones broadcast address (255.255.255.255) while the AIX server was sending to 192.168.0.255.

You may be able to solve this by including a relay between the client and server with the relay configured to use a broadcast of all-ones.

A second option that worked for AIX 5.1 but doesn't seem to work for AIX 5.3 was to:

```
create a host file entry for all-ones (255.255.255.255)
and then add a route:
route add -host all-ones -interface <local-ip-address>
```

The ISC DHCP distribution does not include a dhclient-script for AIX-- AIX comes with a DHCP client. Contribution of a working dhclient-script

for AIX would be welcome.

## MacOS X

The MacOS X system uses a TCP/IP stack derived from FreeBSD with a user-friendly interface named the System Configuration Framework. As it includes a builtin DHCPv4 client (you are better just using that), this text is only about the DHCPv6 client (`dhclient -6 ...`). The DNS configuration (domain search list and name servers' addresses) is managed by a System Configuration agent, not by `/etc/resolv.conf` (which is a link to `/var/run/resolv.conf`, which itself only reflects the internal state; the System Configuration framework's Dynamic Store).

This means that modifying `resolv.conf` directly doesn't have the intended effect, instead the `macos` script sample creates its own `resolv.conf.dhclient6` in `/var/run`, and inserts the contents of this file into the Dynamic Store.

When updating the address configuration the System Configuration framework expects the prefix and a default router along with the configured address. As this extra information is not available via the DHCPv6 protocol the System Configuration framework isn't usable for address configuration, instead `ifconfig` is used directly.

Note the Dynamic Store (from which `/var/run/resolv.conf` is built) is recomputed from scratch when the current location/set is changed. Running the `dhclient-script` reinstalls the `resolv.conf.dhclient6` configuration.

## SUPPORT

The Internet Systems Consortium DHCP server is developed and distributed by ISC in the public trust, thanks to the generous donations of its sponsors. ISC now also offers commercial quality support contracts for ISC DHCP, more information about ISC Support Contracts can be found at the following URL:

<https://www.isc.org/services/support/>

Please understand that we may not respond to support inquiries unless you have a support contract. ISC will continue its practice of always responding to critical items that effect the entire community, and responding to all other requests for support upon ISC's mailing lists on a best-effort basis.

However, ISC DHCP has attracted a fairly sizable following on the Internet, which means that there are a lot of knowledgeable users who may be able to help you if you get stuck. These people generally read the [dhcp-users@isc.org](mailto:dhcp-users@isc.org) mailing list. Be sure to provide as much detail in your query as possible.

If you are going to use ISC DHCP, you should probably subscribe to the `dhcp-users` or `dhcp-announce` mailing lists.

WHERE TO SEND FEATURE REQUESTS: We like to hear your feedback. We may not respond to it all the time, but we do read it. If ISC DHCP doesn't

work well for you, or you have an idea that would improve it for your use, please send your suggestion to [dhcp-suggest@isc.org](mailto:dhcp-suggest@isc.org). This is also an excellent place to send patches that add new features.

WHERE TO REPORT BUGS: If you want the act of sending in a bug report to result in you getting help in the form of a fixed piece of software, you are asking for help. Your bug report is helpful to us, but fundamentally you are making a support request, so please use the addresses described in the previous paragraphs. If you are sure that your problem is a bug, and not user error, or if your bug report includes a patch, you can send it to our ticketing system at [dhcp-bugs@isc.org](mailto:dhcp-bugs@isc.org). If you have not received a notice that the ticket has been resolved, then we're still working on it.

PLEASE DO NOT REPORT BUGS IN OLD SOFTWARE RELEASES! Fetch the latest release and see if the bug is still in that version of the software, and if it is still present, then report it. ISC release versions always have three numbers, for example: 1.2.3. The 'major release' is 1 here, the 'minor release' is 2, and the 'maintenance release' is 3. ISC will accept bug reports against the most recent two major.minor releases: for example, 1.0.0 and 0.9.0, but not 0.8.\* or prior.

PLEASE take a moment to determine where the ISC DHCP distribution that you're using came from. ISC DHCP is sometimes heavily modified by integrators in various operating systems - it's not that we feel that our software is perfect and incapable of having bugs, but rather that it is very frustrating to find out after many days trying to help someone that the sources you're looking at aren't what they're running. When in doubt, please retrieve the source distribution from ISC's web page and install it.

#### HOW TO REPORT BUGS OR REQUEST HELP

When you report bugs or ask for help, please provide us complete information. A list of information we need follows. Please read it carefully, and put all the information you can into your initial bug report. This will save us a great deal of time and more informative bug reports are more likely to get handled more quickly overall.

1. The specific operating system name and version of the machine on which the DHCP server or client is running.
2. The specific operating system name and version of the machine on which the client is running, if you are having trouble getting a client working with the server.
3. If you're running Linux, the version number we care about is the kernel version and maybe the library version, not the distribution version - e.g., while we don't mind knowing that you're running Redhat version mumble.foo, we must know what kernel version you're running, and it helps if you can tell us what version of the C library you're running, although if you don't know that off the top of your head it may be hard for you to figure it out, so don't go crazy trying.
4. The specific version of the DHCP distribution you're running, as reported by `dhcpd -t`.
5. Please explain the problem carefully, thinking through what you're saying to ensure that you don't assume we know

- something about your situation that we don't know.
6. Include your `dhcpd.conf` and `dhcpd.leases` file as MIME attachments if they're not over 100 kilobytes in size each. If they are this large, please make them available to us eg via a hidden <http://> URL or FTP site. If you're not comfortable releasing this information due to sensitive contents, you may encrypt the file to our release signing key, available on our website.
  7. Include a log of your server or client running until it encounters the problem - for example, if you are having trouble getting some client to get an address, restart the server with the `-d` flag and then restart the client, and send us what the server prints. Likewise, with the client, include the output of the client as it fails to get an address or otherwise does the wrong thing. Do not leave out parts of the output that you think aren't interesting.
  8. If the client or server is dumping core, please run the debugger and get a stack trace, and include that in your bug report. For example, if your debugger is `gdb`, do the following:

```
gdb dhcpd dhcpd.core
(gdb) where
      [...]
(gdb) quit
```

This assumes that it's the dhcp server you're debugging, and that the core file is in `dhcpd.core`.

Please see <https://www.isc.org/software/dhcp/> for details on how to subscribe to the ISC DHCP mailing lists.