



NetReg

---

**”An automated DHCP Registration System”**

# **Installing NetReg v1.3rc2 HOWTO**

Instructions by  
**Patrick M. Jaques**  
e-mail: [pjaques@attbi.com](mailto:pjaques@attbi.com)



Last Modified on February 4, 2003

# Table of Contents

<b>Chapter 1. Introduction</b> .....	<b>3</b>
<b>1.1. What is NetReg?</b> .....	<b>3</b>
<b>1.2. How Does NetReg Work?</b> .....	<b>3</b>
<b>1.3. Why use NetReg?</b> .....	<b>3</b>
<b>1.4. Potential pitfalls to NetReg</b> .....	<b>4</b>
<b>1.5. Recommendations</b> .....	<b>4</b>
<b>Chapter 2. NetReg Requirements</b> .....	<b>5</b>
<b>2.1. Pre-requirements</b> .....	<b>5</b>
<b>2.2. Download required software for NetReg</b> .....	<b>5</b>
<b>Chapter 3. Unpack Software Distributions</b> .....	<b>6</b>
<b>Chapter 4. Install Software Distributions</b> .....	<b>7</b>
<b>4.1. Install ISC DHCPd</b> .....	<b>7</b>
<b>4.2. Install Apache 1.3.27 SSL aware</b> .....	<b>7</b>
4.2.1. Build OpenSSL.....	7
4.2.2. Build Mod_SSL.....	7
4.2.3. Build Apache 1.3 .....	7
<b>4.3. Install Perl Module Libnet</b> .....	<b>7</b>
<b>4.4. Install Perl Module Mail::POPClient</b> .....	<b>7</b>
<b>4.5. Install Perl Module Net::IMAP::Simple</b> .....	<b>8</b>
<b>Chapter 5. Configure Software</b> .....	<b>9</b>
<b>5.1. Configure NetReg</b> .....	<b>9</b>
5.1.1. Copy the following files from NetReg distribution.....	9
5.1.2. Rename index.html to register.html .....	9
5.1.3. Create a new index.html .....	10
5.1.4. Configure how NetReg will authenticate your users.....	10
5.1.5. Customize the Default paths that NetReg uses.....	10
5.1.6. Customize register.cgi .....	11
5.1.7. Configure subnets managed by NetReg .....	11
5.1.8. Configure the DHCP lease path in dhcpdctl.....	11
<b>5.2. Configure Apache 1.3.27 w/SSL Support</b> .....	<b>11</b>
5.2.1. Create user/group for Apache/DHCPd & NetReg.....	11
5.2.2. Add ErrorDocument lines to httpd.conf .....	12
5.2.3. Configure Apache httpd.conf restrictions.....	12

5.2.4. Create a .htaccess file .....	13
5.2.5. Create a user Authentication File .....	13
5.2.6. Change ownership/access rights on Apache's directories .....	13
5.2.7. Create a Startup script for Apache's httpd daemon .....	13
5.2.8. Create the proper Symbolic Links for Apache .....	15
<b>5.3. Configuring ISC DHCPd .....</b>	<b>16</b>
5.3.1. Modify the dhcpd.conf .....	16
5.3.2. Copy your dhcpd.conf file .....	17
5.3.3. Change ownership/access rights on DHCP files .....	17
5.3.4. Create a DHCP lease and tmp file .....	17
5.3.5. Create a startup script for the DHCPd daemon .....	18
5.3.6. Create the proper Symbolic Links for ISC DHCPd .....	20
<b>5.4. Configuring DNS Bind .....</b>	<b>20</b>
5.4.1. Configure named.conf .....	20
5.4.2. Configure the DNS Bind cache file .....	21
<b><i>Chapter 6. Scheduling NetReg Update Script .....</i></b>	<b>22</b>
<b><i>Chapter 7. Acknowledgements .....</i></b>	<b>23</b>

# Chapter 1. Introduction

This documentation is meant to serve as a step by step guide in downloading, compiling, installing and configuring a Network Registration system called NetReg on a Red Hat Linux system. Although this documentation is intended for Red Hat 6.0/7.0/8.0 systems, it should work fine on other Linux distributions. While the documentation provides a solid step by step instructional guide on installing NetReg, it does assume that the installer has a basic understanding of Linux, DHCP, DNS Bind and TCP/IP networking.

## 1.1. What is NetReg?

NetReg is an automated network registration system that requires client computers that use DHCP to register their hardware (MAC) address before they can gain full network access.

## 1.2. How Does NetReg Work?

When a client computer running a standard DHCP client is connected to a network, it sends out a DHCP request. When a NetReg DHCP server receives this request, a lookup of the client's MAC address occurs. If the client's computer has previously registered its hardware address, it receives fully functional TCP/IP information (IP address, subnet mask, gateway address, WINS addresses, DNS server addresses), otherwise it receives an IP address within a restrictive network with no access to the internet.

**How is this accomplished?** The NetReg DHCP server is configured with two address pools per subnet. One pool of addresses is assigned to unregistered clients and the other pool is assigned to registered clients. The TCP/IP information passed to unregistered clients has a non-routable IP address or an IP address that is blocked on your firewall, and a bogus DNS server. The bogus DNS server is designed so it resolves all names back to a Network Registration Web Server. When a user starts a web browser, the web browser connects to the NetReg server and redirects all URLs to the NetReg Registration Page. From the Registration Page, the user reads and agrees to your "Acceptable Network Usage Policy", then enters a username/password that is authenticated against a POP/FTP or IMAP server. If the user authenticates successfully, the computer gets registered. After the client reboots, the computer will have full access to the network and the Internet.

## 1.3. Why use NetReg?

DHCP is a standard protocol that automates the process of configuring network hosts by allowing hosts to obtain IP addresses and configuration parameters through the network; DHCP eliminates the need for manual configuration of hosts and manual assignments of IP addresses by network administrators. The problem with DHCP is that there is no security. Any computer can obtain access to your network through a DHCP server and for the most part is essentially anonymous. This makes tracking down malicious users more difficult.

Once a user registers their computer through NetReg's system, it links a user to a computer's hardware (MAC) address. This adds accountability to people's actions while they are connected to your network.

## 1.4. Potential pitfalls to NetReg.

Some clients may learn through others your network configuration and manually configure their computer so they can bypass the NetReg system.

- ❑ Manually assigning DNS servers
- ❑ Manually assigning IP Addresses
- ❑ Manually assigning a Default gateways

### So, what can you do?

- ❑ Look at your switches/router's bridge and/or IP ARP tables and compare them to NetReg's registered hardware (MAC) addresses. This will tell you if you have any rogue users that are statically assigning IP addresses and bypassing your NetReg System. You can accomplish this by scheduling a perl script that uses "Net::SNMP" to periodically check your switches/routers.
- ❑ After identifying any rogue MAC addresses from your switch/router that were not registered by NetReg, your perl script can e-mail this information to your Network Administrator or another account. You may also be able to create a rule on your firewall that bans them from the Internet.

## 1.5. Recommendations

- ❑ Configure NetReg so it uses HTTP connections that run over SSL.
- ❑ Force SSL connections on your NetReg Registration web server. I added a "SSLRequireSSL" directive in my **.htaccess** file. For convenience, I configured NetReg so it redirected index.html to register.html using a SSL connection.

# Chapter 2. NetReg Requirements

## 2.1. Pre-requirements

- ❑ Dedicated PC (Pentium 200MHz, 4GB HD, 32MB RAM, 10/100 Ethernet Adapter)
- ❑ Red Hat Linux 6.0/7.0/8.0, and other Linux platforms
- ❑ DNS Bind 8/9
- ❑ Perl 5 +
- ❑ SSH (recommendation)
- ❑ An FTP or POP server for NetReg to authenticate against
- ❑ Do not install Apache or ISC DHCPd as part of your Linux OS installation or as RPMs later. You will manually compile and install these services later.

## 2.2. Download required software for NetReg

- ❑ Download **netreg-1.3rc2.tar.gz** from <http://www.netreg.org>
- ❑ Download **netreg-cidr.tar.gz** from <http://www.netreg.org/contrib>
- ❑ Download **dhcp-latest.tar.gz** (dhcp-3.0pl2) from <http://www.isc.org>
- ❑ Download **apache\_1.3.27.tar.gz** from <http://www.apache.org>
- ❑ Download **openssl-0.9.6h.tar.gz** from <http://www.openssl.org>
- ❑ Download **mod\_ssl-2.8.12-1.3.27.tar.gz** from <http://www.modssl.org>
- ❑ Download **libnet-1.12.tar.gz** (libnet) from <http://search.cpan.org>
- ❑ Download **POP3Client-2.12.tar.gz** (Mail::POP3Client) from <http://search.cpan.org>
- ❑ Download **Net-IMAP-Simple-0.93.tar.gz** from <http://search.cpan.org/>  
(Net::IMAP::Simple)

# Chapter 3. Unpack Software Distributions

Unpack the following software distributions in /usr/local/src.

- ❑ ***Uncompress apache\_1.3.27.tar.gz***  
cd /usr/local/src  
tar xvzf apache\_1.3.27.tar.gz
- ❑ ***Uncompress dhcp-latest.tar.gz***  
cd /usr/local/src  
tar xvzf dhcp-latest.tar.gz
- ❑ ***Uncompress openssl-0.9.6h.tar.gz***  
cd /usr/local/src  
tar xvzf openssl-0.9.6h.tar.gz
- ❑ ***Uncompress mod\_ssl-2.8.12-1.3.27***  
cd /usr/local/src  
tar xvzf mod\_ssl-2.8.12-1.3.27.tar.gz
- ❑ ***Uncompress libnet-1.12.tar.gz***  
cd /usr/local/src  
tar xvzf libnet-1.12.tar.gz
- ❑ ***Uncompress POP3Client-2.12.tar.gz***  
cd /usr/local/src  
tar xvzf POP3Client-2.12.tar.gz
- ❑ ***Uncompress Net-IMAP-Simple-0.93.tar.gz***  
cd /usr/local/src  
tar xvzf Net-IMAP-Simple-0.93.tar.gz
- ❑ ***Uncompress netreg-1.3rc2.tar.gz***  
cd /usr/local/src  
tar xvzf netreg-1.3rc2.tar.gz
- ❑ ***Uncompress netreg-cidr.tar.gz***  
cd /usr/local/src/netreg-1.3rc2  
tar xvzf netreg-cidr.tar.gz

# Chapter 4. Install Software Distributions

## 4.1. Install ISC DHCPd

```
cd /usr/local/src/dhcp-3.0pl2
./configure
make
make install
```

## 4.2. Install Apache 1.3.27 SSL aware

### 4.2.1. Build OpenSSL

```
cd /usr/local/src/openssl-0.9.6h
./config
make
```

### 4.2.2. Build Mod\_SSL

```
cd /usr/local/src/mod_ssl-2.8.12-1.3.27
./configure --with-apache=../apache_1.3.27 --with-ssl=../openssl-0.9.6h --prefix=/usr/local/apache
```

### 4.2.3. Build Apache 1.3

```
cd /usr/local/src/apache_1.3.27
make
make certificate TYPE=custom
make install
```

## 4.3. Install Perl Module Libnet

```
cd /usr/local/src/libnet-1.12
perl Makefile.PL
make
make test
make install
```

## 4.4. Install Perl Module Mail::POPClient

```
cd /usr/local/src/pop3client-2.12
perl Makefile.PL
make
make test
make install
```



## 4.5. Install Perl Module Net::IMAP::Simple

```
cd /usr/local/src/Net-IMAP-Simple-0.93
perl Makefile.PL
make
make test
make install
```

# Chapter 5. Configure Software

## 5.1. Configure NetReg

### 5.1.1. Copy the following files from NetReg distribution

#### Copied from netreg-1.3rc2.tar.gz

```
cp -r /usr/local/src/netreg-1.3rc2/usr/local/apache/htdocs/ /usr/local/apache
cp -r /usr/local/src/netreg-1.3rc2/usr/local/apache/htdocs/gfx /usr/local/apache/htdocs
cp -r /usr/local/src/netreg-1.3rc2/usr/local/bin/ /usr/local
cp -r /usr/local/src/netreg-1.3rc2/usr/sbin/ /usr
```

#### Copied from netreg-cidr.tar.gz

```
cp -r /usr/local/src/netreg-1.3rc2/usr/local/apache2/cgi-bin/ /usr/local/apache
cp -r /usr/local/src/netreg-1.3rc2/etc/ /
```

- index.html – gets copied to /usr/local/apache/htdocs
- register.cgi – gets copied to /usr/local/apache/cgi-bin
- variables.pl – gets copied to /usr/local/apache/cgi-bin
- subnet.pl – gets copied to /usr/local/apache/cgi-bin
- admin.cgi – gets copied to /usr/local/apache/cgi-bin/admin
- subnet.dat – gets copied to /etc/netreg
- \*.gif – gets copied to /usr/local/apache/htdocs/gfx
- \*.jpg – gets copied to /usr/local/apache/htdocs/gfx
- refresh-dhcpdconf – gets copied to /usr/local/bin
- db.root – gets copied to /etc
- named.conf – gets copied to /etc
- dhcpd.conf – gets copied to /etc/dhcpd
- dhcpdctl - gets copied to /usr/sbin

### 5.1.2. Rename index.html to register.html

```
cd /usr/local/apache/htdocs
mv index.html register.html
```

### 5.1.3. Create a new index.html

So it automatically redirects you to **register.html** using a secure connection. Change the below yellow highlight so it reflects your NetReg server's host name or IP Address.

```
<html>
<head>
  <title> Online Network Registration</title>
  <meta http-equiv="pragma" content="no-cache">
  <meta http-equiv="refresh" content="0; url=https://192.168.0.99/register.html">
</head>
<body>
  Redirecting... please wait. If you are not redirected automatically, then click
  <a href="/register.html"> here</a>.
</body>
</html>
```

### 5.1.4. Configure how NetReg will authenticate your users

Edit **Variables.pl** in **/usr/local/apache/cgi-bin**.

- If you want NetReg to authenticate against a POP server:  
Change **\$AUTH\_METHOD** to **"POP"**.  
Change **\$POPSERVER** to the hostname or IP Address of your mail server.
- If you want NetReg to authenticate against a FTP server:  
Change **\$AUTH\_METHOD** to **"FTP"**.  
Change **\$FTPSERVER** to the hostname or IP Address of your FTP server.
- If you want NetReg to authenticate against a IMAP server:  
Change **\$AUTH\_METHOD** to **"IMAP"**.  
Change **\$IMAPSERVER** to the hostname or IP Address of your IMAP server.

### 5.1.5. Customize the Default paths that NetReg uses

Edit **Variables.pl** in **/usr/local/apache/cgi-bin**. Verify that the following variables are set to the following values. These are the default filename and paths for ISC DHCPd and NetReg 1.3rc2 w/CIDR Update:

```
$LEASESPATH      = "/var/state/dhcp";
$LEASESFILE      = "dhcpd.leases";
$DHCPDCONFPATH  = "/etc/dhcpd";
$DHCPDCONFFILE  = "dhcpd.conf";
$SUBNETFILE      = "/etc/netreg/subnet.dat";
```

### 5.1.6. Customize register.cgi

Edit **register.cgi** in **/usr/local/apache/cgi-bin**. Locate subroutine called 'sub error' and change the HelpDesk message and number.

```
print "Please try to register again. If the problem persists, call the ";  
print "HelpDesk at x6570.<P>";
```

### 5.1.7. Configure subnets managed by NetReg

Edit **subnet.dat** in **/etc/netreg** so it reflects all the subnets on your network that utilize DHCP. The format of subnet.dat is:

- ❑ Subnet/mask length for registered clients
- ❑ Location of subnet (description)
- ❑ Total number of leases from the subnet for registered clients
- ❑ Subnet/mask length for unregistered clients if different, otherwise blank.

Suppose your network consisted of 3 VLANs using a different subnet for registered and unregistered DHCP clients. Registered clients receive a 10.x.x.x IP address and unregistered clients receive a 172.x.x.x IP address. Below is what subnet.dat would resemble:

```
10.1.0.0/16: Dorms Blds: 5100: 172.16.0.0/16:  
10.2.0.0/16: Classrooms: 5100: 172.17.0.0/16:  
10.3.0.0/16: Admin Blds: 5100: 172.18.0.0/16:
```

### 5.1.8. Configure the DHCP lease path in dhcpdctl

Edit NetReg's script called **dhcpdctl** in **/etc/sbin** so the DHCP lease path reflects where ISC DHCP server stores and updates leases.

**Change the DHCPD variable to:**

```
DHCPD='/usr/sbin/dhcpd -cf /etc/dhcpd/dhcpd.conf -lf /var/state/dhcp/dhcpd.leases -q'
```

## 5.2. Configure Apache 1.3.27 w/SSL Support

### 5.2.1. Create user/group for Apache/DHCPd & NetReg

- ❑ Create user & group called webuser with no logon abilities.
- ❑ Edit Apache's httpd.conf file and change the following lines:

**From:**

```
User nobody  
Group nobody
```

**To:**

```
User webuser  
Group webuser
```

## 5.2.2. Add ErrorDocument lines to httpd.conf

```
ErrorDocument 403 /  
ErrorDocument 404 /
```

## 5.2.3. Configure Apache httpd.conf restrictions

Configure Apache's httpd.conf file so your registration server's web directories are properly restricted. The below example restrict access to the registration server so only clients from a 192.168.0.0 network can access it. ( see the below yellow highlights ).

-----Example: portion of httpd.conf-----

```
DocumentRoot "/usr/local/apache/htdocs"  
<Directory />  
    Options FollowSymLinks  
    AllowOverride None  
    Order deny,allow  
    deny from all  
    allow from 192.168.0 127.0.0.1  
</Directory>  
<Directory "/usr/local/apache/htdocs">  
    Options Indexes FollowSymLinks MultiViews  
    AllowOverride None  
    Order deny,allow  
    deny from all  
    allow from 192.168.0 127.0.0.1  
</Directory>  
AccessFileName .htaccess  
<Files ~ "\.ht">  
    Order allow,deny  
    Deny from all  
    Satisfy All  
</Files>  
Alias /icons/ "/usr/local/apache/icons/"  
<Directory "/usr/local/apache/icons">  
    Options Indexes MultiViews  
    AllowOverride None  
    Order deny,allow  
    deny from all  
    allow from 192.168.0 127.0.0.1  
</Directory>  
ScriptAlias /cgi-bin/ "/usr/local/apache/cgi-bin/"  
<Directory "/usr/local/apache/cgi-bin">  
    AllowOverride AuthConfig  
    Options None  
    Order deny,allow
```

```
deny from all
allow from 192.168.0 127.0.0.1
</Directory>
```

#### 5.2.4. Create a .htaccess file

Place this file ( `.htaccess` ) in your `/usr/local/apache/cgi-bin/admin` to protect NetReg's administrative script:

```
AuthUserFile /usr/local/apache/conf/htusers
AuthName "Network Registration"
AuthType Basic
AuthAuthoritative on
SSLRequireSSL
Require valid-user
```

#### 5.2.5. Create a user Authentication File

Create a user authentication file called 'htusers' and place it in `/usr/local/apache/conf`.

Ex. `/usr/local/apache/bin/htpasswd -c /usr/local/apache/conf/htusers netreg`

The `-c` argument tells `htpasswd` to create a new users file for Apache in the specified directory. When you run this command, you will be prompted to enter a password for 'netreg', and then confirm it. Other users can be added to the existing file in the same way without the `-c` argument.

FYI: To allow a directory to be restricted within a `.htaccess` file, you first need to ensure that the `httpd.conf` allows user authentication to be set up in a `.htaccess` file. This is controlled by the `AuthConfig` override. The `httpd.conf` file should include `AllowOverride AuthConfig` under the `<Directory "/usr/local/apache/cgi-bin">` directive.

#### 5.2.6. Change ownership/access rights on Apache's directories

```
chown -R root:webuser /usr/local/apache
```

#### 5.2.7. Create a Startup script for Apache's httpd daemon

Since you are manually compiling and installing Apache from scratch (not using a RPM), you will need to create a startup script called '`httpd`' to start Apache when you boot your server. Place the startup script '`httpd`' in the `/etc/rc.d/init.d` directory. After creating a startup script for Apache, change the file permissions of your startup script so the Owner has read/write/execute access, Group has read/execute access and Other has read/execute access.

Example: `chmod 755 /etc/rc.d/init.d/httpd`

**Make sure you start Apache's http daemon with SSL support by using either:**

```
/usr/local/apache/bin/apachectl startssl (or)
/usr/local/apache/bin/httpd -DSSL
```

-----Example: Apache Script -- (Put in /etc/rc.d/init.d)-----

```
#!/bin/bash
# httpd - init file for Apache
#
# description: Apache Web Server protocol (HTTP) Daemon
#
# processname: /usr/local/apache/bin/httpd
# config: /usr/local/apache/conf/httpd.conf
# pidfile: /var/run/httpd
#
# source function library
. /etc/init.d/functions
#
OPTIONS="startssl"
RETVAL=0
prog="httpd"

start() {
    echo -n $"Starting $prog: "
    daemon /usr/local/apache/bin/apachectl $OPTIONS
    RETVAL=$?
    echo
    touch /var/lock/subsys/httpd
    return $RETVAL
}

stop() {
    echo -n $"Stopping $prog: "
    killproc /usr/local/apache/bin/httpd
    RETVAL=$?
    echo
    rm -f /var/lock/subsys/httpd
    return $RETVAL
}

reload(){
    echo -n $"Reloading $prog: "
    killproc /usr/local/apache/bin/httpd -HUP
    RETVAL=$?
    echo
    return $RETVAL
}

restart(){
    stop
    start
}

condrestart(){
    [ -e /var/lock/subsys/httpd ] && restart
```

```

        return 0
    }
    case "$1" in
        start)
            start
            ;;
        stop)
            stop
            ;;
        restart)
            restart
            ;;
        reload)
            reload
            ;;
        condrestart)
            condrestart
            ;;
        status)
            status httpd
            RETVAL=$?
            ;;
        *)
            echo $"Usage: $0 {start|stop|status|restart|condrestart|reload}"
            RETVAL=1
    esac
    exit $RETVAL

```

### 5.2.8. Create the proper Symbolic Links for Apache

You will need to create symbolic links to the 'httpd' startup script in the run-level directories (rc0.d, rc1.d, rc2.d, rc3.d, rc4.d, rc5.d, rc6.d) in order for Apache to be automatically started and shutdown properly.

#### Create the proper Symbolic Links

```

cd /etc/rc.d/init.d
ln -s httpd /etc/rc.d/rc0.d/K37httpd
ln -s httpd /etc/rc.d/rc1.d/K37httpd
ln -s httpd /etc/rc.d/rc2.d/S45httpd
ln -s httpd /etc/rc.d/rc3.d/S45httpd
ln -s httpd /etc/rc.d/rc4.d/S45httpd
ln -s httpd /etc/rc.d/rc5.d/S45httpd
ln -s httpd /etc/rc.d/rc6.d/K37httpd

```



## 5.3. Configuring ISC DHCPd

### 5.3.1. Modify the dhcpd.conf.

Modify the **dhcpd.conf** file in **/etc/netreg** so it reflects your network infrastructure. The below example dhcpd.conf file provides DHCP support to one network, which is 192.168.0.0, although it could support many networks/subnets and shared networks.

-----Example dhcpd.conf—(single network)-----

```
# dhcpd.conf
# Configuration file for ISC dhcpd
# Option definitions common to all supported networks...
option domain-name "homedomain.cc";
option netbios-node-type 8;
server-identifier mrhanky.homedomain.cc;
max-lease-time 120;
default-lease-time 120;
allow bootp;
allow booting;
ddns-update-style ad-hoc;
ddns-updates off;
omapi-port 7911;

# Define subnet pools for known/unknown clients
subnet 192.168.0.0 netmask 255.255.255.0 {
    authoritative;
    option subnet-mask 255.255.255.0;
    option broadcast-address 192.168.0.255;
    pool {
        range 192.168.0.100 192.168.0.150;
        option domain-name-servers 192.168.0.99;
        default-lease-time 120;
        max-lease-time 120;
        allow unknown clients;
    }
    pool {
        range 192.168.0.151 192.168.0.200;
        option routers 192.168.0.254;
        option domain-name-servers 24.218.0.228, 24.218.0.229;
        default-lease-time 1209600;
        max-lease-time 1209600;
        deny unknown clients;
    }
}
```

**Note:** A shared-network directive in the dhcpd.conf file allows ISC DHCP server to support multiple networks to on the same wire. This of course means that your router or Layer 3 switch is configured with overlapping VLANs.

-----Example portion of dhcpd.conf—(shared-network)-----

```
shared-network campus {
    subnet 192.168.1.0 netmask 255.255.255.0 {
        option broadcast-address 192.168.1.255;
        option routers 192.168.1.254;
        range 192.168.1.50 192.168.1.200;
    }
    subnet 192.168.2.0 netmask 255.255.255.0 {
        option broadcast-address 192.168.2.255;
        option routers 192.168.2.254;
        range 192.168.2.50 192.168.2.240;
    }
}
```

### 5.3.2. Copy your dhcpd.conf file.

In order for NetReg's system to update the dhcp registrations, you will need to make copies of dhcpd.conf.

```
# cd /etc/dhcpd
# cp -f dhcpd.conf dhcpd.conf.new
# cp -f dhcpd.conf dhcpd.conf.bak
```

When a client register's their computer's hardware address through NetReg's registration web server, it puts the hardware address in 'dhcpd.conf.new'. NetReg's scripts check every minute whether 'dhcpd.conf.new' has been updated and if so, it stop's the dhcpd service, copies 'dhcpd.conf.new' to 'dhcpd.conf' and then restarts the dhcpd service.

### 5.3.3. Change ownership/access rights on DHCP files

Change the ownership/access rights to ISC DHCP configuration files in /etc/dhcpd.

```
# chown -R root:webuser /etc/dhcpd
# chmod 664 /etc/dhcpd/*
```

### 5.3.4. Create a DHCP lease and tmp file

Create a lease file and tmp file in /var/state/dhcp.

```
# touch /var/state/dhcp/dhcpd.leases
# touch /var/state/dhcp/dhcpd.leases.tmp
```

### 5.3.5. Create a startup script for the DHCPd daemon

Since you are manually compiling and installing ISC's dhcpd from scratch (not using a RPM), you will need to create a startup script called '**dhcpd**' to start ISC's dhcpd daemon when you boot your server. Place the startup script '**dhcpd**' in the **/etc/rc.d/init.d** directory. After creating a startup script for dhcpd, change the file permissions of your startup script so the Owner has read/write/execute access, Group has read/execute access and Other has read/execute access.

Example: **chmod 755 /etc/rc.d/init.d/dhcpd**

-----Example Script to Start dhcpd-----

```
#!/bin/bash
# dhcpd - init file for ISC DHCPd
#
# description: Internet Software Consortium DHCP Server Daemon

# source function library
. /etc/init.d/functions

# source networking functions
. /etc/sysconfig/network

# Check that networking is up.
[ ${NETWORKING} = "no" ] && exit 0

# Check that ISC DHCPD files exist
[ -f /usr/sbin/dhcpd ] || exit 0
[ -f /etc/dhcpd/dhcpd.conf ] || exit 0
[ -f /var/state/dhcp/dhcpd.leases ] || exit 0

OPTIONS="-cf /etc/dhcpd/dhcpd.conf -lf /var/state/dhcp/dhcpd.leases"
RETVAL=0
prog="dhcpd"

start() {
    echo -n $"Starting $prog: "
    daemon /usr/sbin/dhcpd $OPTIONS
    RETVAL=$?
    echo
    touch /var/lock/subsys/dhcpd
    return $RETVAL
}

stop() {
    echo -n $"Stopping $prog: "
    killproc /usr/sbin/dhcpd
    RETVAL=$?
}
```

```

        echo
        rm -f /var/lock/subsys/dhcpd
        return $RETVAL
    }
    reload(){
        echo -n $"Reloading $prog: "
        killproc /usr/sbin/dhcpd -HUP
        RETVAL=$?
        echo
        return $RETVAL
    }
    restart(){
        stop
        start
    }
    condrestart(){
        [ -e /var/lock/subsys/dhcpd ] && restart
        return 0
    }
    case "$1" in
        start)
            start
            ;;
        stop)
            stop
            ;;
        restart)
            restart
            ;;
        reload)
            reload
            ;;
        condrestart)
            condrestart
            ;;
        status)
            status dhcpd
            RETVAL=$?
            ;;
        *)
            echo $"Usage: $0 {start|stop|status|restart|condrestart|reload}"
            RETVAL=1
    esac
    exit $RETVAL

```

### 5.3.6. Create the proper Symbolic Links for ISC DHCPd

You will need to create symbolic links to the 'dhcpd' startup script in the run-level directories (rc0.d, rc1.d, rc2.d, rc3.d, rc4.d, rc5.d, rc6.d) in order for ISC DHCPd to be automatically started and shutdown properly.

#### Create the proper Symbolic Links:

```
cd /etc/rc.d/init.d
ln -s dhcpd /etc/rc.d/rc0.d/K38dhcpd
ln -s dhcpd /etc/rc.d/rc1.d/K38dhcpd
ln -s dhcpd /etc/rc.d/rc2.d/S47dhcpd
ln -s dhcpd /etc/rc.d/rc3.d/S47dhcpd
ln -s dhcpd /etc/rc.d/rc4.d/S47dhcpd
ln -s dhcpd /etc/rc.d/rc5.d/S47dhcpd
```

## 5.4. Configuring DNS Bind

### 5.4.1. Configure named.conf

Edit **named.conf** in **/etc**. Change the server directive so it reflects your NetReg server's hostname or IP address.

-----Example named.conf-----

```
// named.conf for NetReg
//
server 192.168.0.99 {
    bogus yes;
};
options {
    directory "/etc/";
    recursion no;
    fetch-glue no;
};
zone "." in {
    type master;
    file "db.root";
};
```

## 5.4.2. Configure the DNS Bind cache file

Edit /etc/db.root.

- Add **\$TTL 3600**
- Change hostname and domain in **SOA** and **NS** lines.
- Change IP address and hostname in **A** lines.

-----Example db.root-----

**; Bind 8 -- Zone file -- for NetReg**

**\$TTL 3600**

```
. IN SOA mrhanky.homedomain.cc. root.mrhanky.homedomain.cc. (  
    2 ; serial  
    10800 ; refresh  
    3600 ; retry  
    604800 ; expire  
    86400 ; default_ttl )  
    IN NS mrhanky.homedomain.cc.  
mrhanky 86400 IN A 192.168.0.99  
*. 86400 IN A 192.168.0.99
```

# Chapter 6. Scheduling NetReg Update Script

Schedule the **refresh-dhcpdconf** script so it runs every minute.

Since the DHCP server only reads its conf file when it is started, you will need a cron job that checks to see if the server needs to re-read its configuration. The **refresh-dhcpdconf** script helps do this but cron will need to run this script every minute or two. We have cron running the script every minute as root like so in `/var/spool/cron/root`. FYI: Don't edit `/var/spool/cron/root` directly, use `crontab -e`.

```
0-59/1 * * * * /usr/local/bin/refresh-dhcpdconf
```

Within a minute, **refresh-dhcpdconf** will check if the `dhcpd.conf.new` has been updated. If `dhcpd.conf.new` is newer than `dhcpd.conf`, it updates `dhcpd.conf` and restart the DHCP server.

- You have now successfully built a Network Registration server utilizing NetReg v1.3 rc2 and CIDR update. Please reboot your server or manually start the '**dhcpd**' and '**httpd**' services.

```
/etc/rc.d/init.d/dhcpd start    or    service dhcpd start  
/etc/rc.d/init.d/httpd start   or    service httpd start
```

# Chapter 7. Acknowledgements

Many thanks go to the developers and contributors of the NetReg project. Their dedication has helped many network administrators sleep soundly at night in that we can all take solace in knowing who are using our networks.

## **Programming:**

Peter Valian [valianp@southwestern.edu](mailto:valianp@southwestern.edu)

## **Documentation:**

Todd K Watson [tkw@southwestern.edu](mailto:tkw@southwestern.edu)

Peter Valian [valianp@southwestern.edu](mailto:valianp@southwestern.edu)

## **Code Contributors:**

Allie M. Tate [allie@lsu.edu](mailto:allie@lsu.edu)

Bobby Clark [bobby.clark@eku.edu](mailto:bobby.clark@eku.edu)

Robert Lowe [robert.h.lowe@lawrence.edu](mailto:robert.h.lowe@lawrence.edu)