

First things first

You're going to need 2 network cards. Take your first network card, and plug your WAN connection into it. You should know what network card this is, eth0 eth1 ect... If you don't know what it is, trial and error my friend.

Let's just say that your WAN card is going to be eth0 (*if it's eth1, just do everything the same but adjust your config accordingly*). We want to locate the file /etc/network/interfaces. Do a VI on the file such as

```
sudo vi /etc/network/interfaces
```

You should see in the file (*if you have nothing setup yet*):

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
# The loopback network interface
auto lo
iface lo inet loopback
```

We are going to add this to the file. *As a side note, if you don't know how to use VI use nano or learn VI.*

```
auto eth0
iface eth0 inet dhcp
```

The auto eth0 code tells eth0 to start on boot, similar to running

```
sudo ifconfig eth0 up
```

The code iface eth0 inet dhcp tells the eth0 interface to look for a DHCP server and get its info from there. This is important if you're hooked up to a cable modem, as you will want to get a public IP from your ISP.

The next step to take is to configure your network card eth1. This will be your "LAN" card.

If you remember, our /etc/network/interfaces configuration looked like

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
# The loopback network interface
auto lo
iface lo inet loopback
# The primary network interface
auto eth0
iface eth0 inet dhcp
```

We are going to VI into the interfaces file again and add a few more lines:

```
sudo vi /etc/network/interfaces
```

Add these lines to the bottom of the file.

```
auto eth1
```

```
iface eth1 inet static
    address      172.17.207.121
    netmask      255.255.0.0
    broadcast    172.17.207.255
    network      172.17.207.0
```

This just gives you a static IP address for your server on your LAN card.

Your file should now look like this.

```
# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).
# The loopback network interface
auto lo
iface lo inet loopback
# The primary network interface
auto eth0
iface eth0 inet dhcp
auto eth1
iface eth1 inet static
    address      172.17.207.121
    netmask      255.255.0.0
    broadcast    172.17.207.255
    network      172.17.207.0
```

Now, before we forget, let's edit your /etc/hosts file.

```
sudo vi /etc/hosts
```

Make the file look like mine, though if you call your server `userve` or `myserver` you can change it.

Also note the `asus.local` domain name, it's a good idea to use your own such as `mydomain.local` but you can use what I have for learning purposes.

We use `.local` because it's easy to remember and it's not public, so we will not interfere with anything.

```
127.0.0.1      localhost server.localhost
172.17.207.121 server.asus.local server asus.local
```

Now that we have our interfaces configured, we are going to install and set up a dhcp server. To install the dhcp server run the command

```
sudo apt-get install dhcp3-server
```

Let's edit the `dhcpd.conf` file. Start by running the command

```
sudo vi /etc/dhcp3/dhcpd.conf
```

Now if there is anything in that file, **REMOVE IT**.

Copy and paste this into your file, then write and quit.

```
ddns-update-style none;
option domain-name "whatever.local"; //change this to something you want.local
such as mydomain.local
option domain-name-servers 172.17.207.121, 24.92.226.41; //you also might want
to change that second dns server to your ISP's local DNS server
option routers 172.17.207.121;
default-lease-time 42300;
max-lease-time 84600;
authoritative;
log-facility local7;
subnet 172.17.0.0 netmask 255.255.0.0 {
    range 172.17.207.1 172.17.207.100; //you can expand the range just by
changing .100 to .254 or something like that
}
```

Now run the command

```
sudo /etc/init.d/dhcp3-server start
```

This will start your DHCP server and we can label this part DONE.

Moving on to... DNS

Bind is the DNS package that we will be using. To install this, we just simply run

```
sudo apt-get install bind9
```

This will download and install our bind server.

Start by running the command

```
vi /etc/bind/named.conf
```

Then remove everything in the file and look for my comments, usually indicated by //.

```
// This is the primary configuration file for the BIND DNS server named.
//
// Please read /usr/share/doc/bind9/README.Debian.gz for information on the
// structure of BIND configuration files in Debian, *BEFORE* you customize
// this configuration file.
//
// If you are just adding zones, please do that in /etc/bind/named.conf.local
include "/etc/bind/named.conf.options";
// prime the server with knowledge of the root servers
zone "." {
    type hint;
    file "/etc/bind/db.root";
};
// be authoritative for the localhost forward and reverse zones, and for
// broadcast zones as per RFC 1912
zone "asus.local" { //change asus.local to whatever you named your domain such
as mydomain.local
type master;
```

```

file "/etc/bind/zones/asus.local.db"; //this file or folder does not exist so we
will need to make it
};
zone "207.17.172.in-addr.arpa" {
type master;
file "/etc/bind/zones/rev.207.17.172.in-addr.arpa";//this file does not exist so
we will also need to make it
};
zone "localhost" {
    type master;
    file "/etc/bind/db.local";
};
zone "127.in-addr.arpa" {
    type master;
    file "/etc/bind/db.127";
};
zone "0.in-addr.arpa" {
    type master;
    file "/etc/bind/db.0";
};
zone "255.in-addr.arpa" {
    type master;
    file "/etc/bind/db.255";
};
include "/etc/bind/named.conf.local";

```

Before we can make the two files asus.local.db and rev.207.17.172.in-addr.arpa, we need to edit another file. So

```
sudo vi /etc/bind/named.conf.options
```

Remove everything in the file and use this...

```

options {
    directory "/var/cache/bind";
    // If there is a firewall between you and nameservers you want
    // to talk to, you may need to fix the firewall to allow multiple
    // ports to talk. See http://www.kb.cert.org/vuls/id/800113
    // If your ISP provided one or more IP addresses for stable
    // nameservers, you probably want to use them as forwarders.
    // Uncomment the following block, and insert the addresses replacing
    // the all-0's placeholder.
    forwarders {
        24.92.226.41; //very important, change this to your LOCAL ISP's DNS
server(s)
        24.92.224.40;
    };
    auth-nxdomain no;    # conform to RFC1035
    listen-on-v6 { any; };
};

```

Pay attention to the comments, they tell you to **CHANGE** our forwarders address(es) to your LOCAL ISP's DNS.

Next, cd over to your bind directory:

```
cd /etc/bind/  
sudo mkdir zones  
cd zones  
sudo vi asus.local.db
```

(Or use your domain name such as mydomain.local.db.)

Once you are in the asus.local.db file or mydomain.local.db file (whatever you called it), copy and paste this, make the appropriate changes to your domain name.

```
$ORIGIN .  
$TTL 4000 ;  
asus.local.      IN SOA  server.asus.local. admin.asus.local. (  
2007031001      ; serial  
28800           ; refresh  
3600            ; retry  
604800          ; expire  
38400           ; min  
)  
                NS      server.asus.local.  
$ORIGIN asus.local.  
                IN      A      172.17.207.121  
www             IN      A      172.17.207.121 //an example  
server         IN      A      172.17.207.121 //an example  
macpro         IN      A      172.17.207.4   //an example
```

If you do an nslookup macpro, you will get 172.17.207.4 back as an answer, so change the domain names and IP's according to your settings.

Next, we are going to vi the rev.207.17.172.in-addr.arpa file that does not exist yet. But it will once we save it. So assuming you're still in the zones folder:

```
vi rev.207.17.172.in-addr.arpa
```

Copy and paste what I have here, making the appropriate changes.

```
$ORIGIN .  
$TTL 28800      ; 8 hours  
207.17.172.IN-ADDR.ARPA IN SOA  server.asus.local. admin.asus.local. (  
2008110601     ; serial  
28800          ; refresh (8 hours)  
7200           ; retry (2 hours)  
604800         ; expire (1 week)  
86400          ; minimum (1 day)  
)  
                NS      server.asus.local.  
$ORIGIN 207.17.172.IN-ADDR.ARPA.  
4             PTR      macpro.asus.local.
```

So now if you did a reverse lookup on 172.17.207.4, you would get macpro.asus.local.

Now run the command to start named:

```
sudo /etc/init.d/named start
```

If it does not start, check the logs in /var/logs.

Last but not least, IPTABLES

First thing is first, we need to edit sysctl.conf in the folder /etc/, so:

```
sudo vi /etc/sysctl.conf
```

Uncomment line 28. That means removing the # in front of it. The line should be net.ipv4.ip_forward=1

Next, let's vi over to rc.local:

```
sudo vi /etc/rc.local
```

Add these two lines to the bottom of the file:

```
/sbin/iptables -P FORWARD ACCEPT  
/sbin/iptables --table nat -A POSTROUTING -o eth0 -j MASQUERADE
```

This will set up your gateway using iptables. You can use iptables to make this more secure than this basic setup.

To forward ports, you can add something like this to the end of the rc.local file.

```
/sbin/iptables -t nat -A PREROUTING -p tcp -i eth0 -d jgibbs.dyndns.org --dport  
3389 -j DNAT --to 172.17.207.4:3389
```

The long line above will port forward all incoming traffic on port 3389 to the IP 172.17.207.4, so I can remote desktop into my Windows box from outside my network.

You can do this with any ports you wish.

RESTART!

Also, report any problems and I will fix this tutorial with updates.