

# Lab 6.1 Setting up BIND on OpenBSD

## IN715 Networks Three

August 26, 2014

### Introduction

OpenBSD is an excellent platform on which to run BIND. It provides the sort of stability, network performance, and security that are highly desirable when running core infrastructure services. BIND is installed as a base package and its default configuration is sensible. We do need to do some additional configuration before we are ready to run BIND. We will perform that configuration in this lab.

This lab assumes that basic networking is properly configured on your OpenBSD server.

### 1 Goals

We want to configure BIND to perform the following tasks:

1. Respond to recursive queries from clients on our networks;
2. Serve as a master authoritative servers for our zone `nw3.sqrawler.com`;
3. Provide reverse lookups for our networks.

### 2 Important files

OpenBSD runs `named`, the BIND DNS service, in a *chroot* environment located at `/var/named`. Most of the files we will use are under this directory tree. In particular we will examine

- `etc/named.conf`, the primary `named` configuration file;
- `etc/root.hints`, the file containing root server addresses;
- `master`, the directory where we will place our zone files.

In addition to these, we will need to edit `/etc/rc.conf.local` to set flags allowing `named` to start. We will use, but not edit, `/etc/rc.d/named` to start and stop the service.

### 3 Responding to recursive queries

Our server should respond to *recursive queries* from local clients, but not to other clients. It turns out that it is nearly configured to do this now. Open `/var/named/etc/named.conf` with a text editor to see how.

First, note the following *access control list* (acl):

```
acl clients {  
    localnets;  
    ::1;  
}
```

and the following line from the options section:

```
allow-recursion clients; ;
```

The allow-recursion directive means that `named` will respond to recursive DNS queries from clients identified in the “clients” acl. That acl includes `localnets`, which is shorthand for any directly connected networks, and for the IPv6 local host address. We just need to add a line to the acl for our other internal network that is not directly connected to the server. The line looks like this:

```
192.168.2/24;
```

## 4 Starting named

Before we can start `named`, we need to add a line to `/etc/rc.conf.local` that says

```
named_flags=""
```

Then we can start `named` with the command

```
/etc/rc.d/named start
```

## 5 Configuring our local resolver

Our client machines get their DNS server addresses from DHCP, but we need to set it ourselves for our BSD server by editing the file `/etc/resolve.conf`. Remove any current lines and add the following:

```
search nw3.sqrawler.com  
nameserver 127.0.0.1
```

The first line says that hostnames without a domain name will be assumed to be in the `nw3.sqrawler.com` domain. The second says that our BSD server will use itself for DNS queries.

Use the `dig` command on your BSD server to be sure that it is responding to DNS queries. For example, try

```
dig a www.trademe.co.nz
```

and see that you get an address.

## 6 Adding zone files

In the last lab we prepared two *zone files* for use with BIND. Place those files in `/var/named/master`. Be sure to note the file names you used.

Now we need to add the zone entries to `/var/named/etc/named.conf`. Add an entry for the `nw3.sqrawler.com` zone as follows.

```
zone "nw3.sqrawler.com" {  
    type master;  
    file "master/db.nw3.sqrawler.com";  
    allow-transfer { localnets; };  
};
```

The `type` line says that we are a master server for this zone rather than a slave. The `file` line indicates where the zone file is. The `allow-transfer` line says that we will transfer zone data to slaves on our local networks only.

Create similar sections for your reverse zone, restart the server, and then use `dig` to be sure your DNS is functioning properly.