Setting up a caching only DNS server

There are many software options for DNS, but the one that is almost universally used in the POSIX world is BIND (Berkeley Internet Name Domain). DNS is how your system knows what ip address a request needs to be sent to when you provide a hostname. Complexity will depend upon your needs.

Service Name:
named

Package:
bind

Config:
/etc/named.conf

Access Control:
iptables
SELinux
Built in access control

Log Files:
/var/log/messages

BIND is a service with the purpose of resolving hostnames to ip addresses and vice versa. This is accomplished by a hierarchy of servers that have authoritative records for every namespace that can be resolved. These records look like:

google.com.	34	IN	A	74.125.227.110
Domain name	Priority	Class	Record type	IP address associated with this record.

For simplicity's sake we will discuss a type of DNS server that holds no records of it's own. This kind of DNS server is known as a "caching only nameserver."

The purpose of a caching only nameserver is to minimize searches to pages that have already been resolved by somebody else using that server within what is known as the "Time To Live" or TTL. The clock for the TTL starts when the caching only nameserver retrieves a record to fill a query from a client. That record is cached for the duration of the TTL so that the result can be returned immediately to other clients.

To install:

```
# yum -y install bind
```

Assuming that your nameserver (where you resolve names from) is 192.168.0.254, make the following entry in /etc/named.conf:

```
listen-on port 53 { any; };
allow-query {192.168.0.0/24; };
forwarders { 192.168.0.254; };
```

This will allow queries from your own network (192.168.0.0/24) and forward all requests that could not be resolved internally to your nameserver.

Start the service persistently.

```
# service named start
# chkconfig named on
```

Here is a quick command to see if the syntax of your config file is acceptable (you will only need to run this command if named will not start properly on the previous step):

```
# service named configtest
```

It really is that easy.

Create firewall rules to allow connectivity over the network to your DNS server:

```
# iptables -I INPUT -p udp --dport 53 -j ACCEPT
# service iptables save
```

Adapt the /etc/named.conf to your needs:

Important parameters:

```
listen-on
allow-query
```

forwarders

listen-on defines the UDP port and the interface the named is bound to. By default named is bound to localhost only so it won't service requests from the network.

allow-query defines which IP addresses or IP networks are allowed to use this DNS server. By default only localhost can invoke queries.

forwarders defines which DNS server(s) will handle queries which are not in your DNS server's cache. This parameter does not exist by default.

Notes:

- Parameters have their values enclosed in { }'s
- For every value assigned to parameter needs to be closed with a ;
- Each line is terminated with a ;
- Lines which begin with a # or // or /* are considered comments
- Don't forget to run service named restart after editing /etc/named.conf

Examples:

```
options {
        listen-on port 53 { 127.0.0.1; 192.168.122.3; };
        listen-on-v6 port 53 { ::1; };
        directory       "/var/named";
        dump-file       "/var/named/data/cache_dump.db";
        statistics-file "/var/named/data/named_stats.txt";
        memstatistics-file "/var/named/data/named_mem_stats.txt";
        allow-query    { 192.168.122.0/24; };
        forwarders { 8.8.8.8; };
        recursion yes;

        dnssec-enable yes;
        dnssec-validation yes;
        dnssec-lookaside auto;

        /* Path to ISC DLV key */
        bindkeys-file "/etc/named.iscdlv.key";

        managed-keys-directory "/var/named/dynamic";
};
```

Explanation:

The DNS server is listening on UDP port 53 and is available on IP addresses 127.0.0.1 and 192.168.122.3

Clients on the 192.168.122.0/24 network can query this DNS server

Should a client request resolution that is not in the DNS server's cache, it will be passed onto the upstream DNS server with an IP address of 8.8.8.8

Lab activity

Make sure that your server forwards all DNS queries to 8.8.4.4 for the IP network 192.168.122.0/24